

HIGHLY ACCURATE VISION BASED SURFACE  
RECONSTRUCTION SYSTEM

TAUFIQUR RAHMAN









# Highly Accurate Vision Based Surface Reconstruction System

by

© Taufiqur Rahman

A thesis submitted to the  
School of Graduate Studies  
in partial fulfilment of the  
requirements for the degree of  
Master of Engineering

Faculty of Engineering & Applied Science  
Memorial University of Newfoundland

July 2009

St. John's

Newfoundland

To my mother

## Abstract

Photogrammetry has come a long way since cameras were employed for aerial mapping tasks. In this digital era, photogrammetry evolved to be used in close range measurements for industrial purposes, thanks to inexpensive and off-the-shelf digital cameras and superior computing power. But how accurately close range photogrammetry can execute a measurement is a question still unexplored by the researchers. This thesis attempts this question to find a vision based measurement solution that employs conventional sensing means yet produces highly accurate results. In a way, the proverbial "pushing the limit" was the main focus of this work.

It was found that geometrical calibration of digital cameras is the major task in developing a vision based measurement system. This work proposes a novel calibration technique with an improved approach towards implementation. Experimental results and computer simulation confirmed the accuracy produced by the proposed camera calibration technique. A laser projector emitting a linear pattern was used as an additional sensing element in complement to the camera. Finally a spherical surface was digitized in order to assess the performance of the overall system.

## Acknowledgements

I would like to express my deepest appreciation for my supervising professor Dr. Nicholas Krouglicof who throughout this work offered his guidance and conveyed a spirit of adventure in regard to this research. I also thankfully acknowledge his contributions in setting up the calibration rig. Without his persistent help and encouragement this thesis would not have been possible.

I extend my deep appreciation to the examiners of this thesis. Their diligent work contributed heavily to the successful completion of this thesis.

I gratefully thank my family for being the constant source of inspiration and for being so understanding when I couldn't be there because of my work. The journey would not have been so smooth if it wasn't for them.



# Contents

<b>Abstract</b>	<b>i</b>
<b>Acknowledgements</b>	<b>ii</b>
<b>List of Tables</b>	<b>viii</b>
<b>List of Figures</b>	<b>x</b>
<b>1 Introduction</b>	<b>1</b>
1.1 Problem Definition . . . . .	2
1.2 Challenges . . . . .	3
1.3 System Architecture . . . . .	4
1.4 Organization of thesis . . . . .	5
<b>2 Camera Model: Mapping of 3D to 2D</b>	<b>8</b>
2.1 Nomenclature . . . . .	9
2.2 Pinhole Camera Model . . . . .	9
2.3 Modeling Image Formation . . . . .	11
2.4 Forward Camera Model . . . . .	12

2.4.1	Transformation from World Coordinate Frame to Camera Coordinate Frame . . . . .	13
2.4.2	Perspective Projection Transformation from Camera Coordinate Frame to Retinal Coordinate Frame . . . . .	14
2.4.3	Transformation from Retinal Coordinate Frame to Image Coordinate Frame . . . . .	16
2.4.4	From Undistorted Image Coordinates to Distorted Image Coordinates . . . . .	17
2.4.5	Overall Transformation . . . . .	20
2.5	Camera Parameters . . . . .	21
<b>3</b>	<b>Camera Calibration Methods: A Review</b>	<b>23</b>
3.1	Classification of Camera Calibration Methods . . . . .	24
3.2	Camera Calibration Techniques in Computer Vision . . . . .	27
3.3	Experimental Evaluation of Calibration Techniques . . . . .	29
<b>4</b>	<b>Propositions for Improvement in Methodology</b>	<b>32</b>
4.1	Representation of Orientation . . . . .	33
4.2	Unit Quaternions: Non-singular Representation of Rotation . . . . .	35
4.2.1	Quaternion Algebra . . . . .	35
4.2.2	Conjugate, Norm, Inverse . . . . .	37
4.2.3	Rotation Representation . . . . .	38
4.2.4	In Context of the Camera Model . . . . .	45
4.3	Modeling Lens Distortion . . . . .	46
4.4	Numerical Solution to the Calibration Problem . . . . .	50

4.5	Choice of the Error Criterion . . . . .	51
<b>5</b>	<b>An Accurate Camera Calibration Technique</b>	<b>53</b>
5.1	Formulation of the Calibration Problem . . . . .	53
5.2	Initial Approximation . . . . .	63
5.2.1	Estimating Extrinsic Parameters . . . . .	65
5.2.2	Linear Least Squares Analysis . . . . .	65
5.2.3	Forcing Orthonormality . . . . .	70
5.2.4	Estimating the Rotation Matrix . . . . .	70
5.2.5	Retrieving the Range of the Target . . . . .	71
5.2.6	Sign Ambiguity . . . . .	71
5.3	Calibration Algorithm . . . . .	72
5.4	Implementation of the Algorithm . . . . .	75
<b>6</b>	<b>Implementation of the Camera Calibration Technique</b>	<b>77</b>
6.1	Conventional Calibration Targets . . . . .	78
6.1.1	Accuracy in the Image Space . . . . .	79
6.1.2	Accuracy in the Object Space . . . . .	82
6.2	Generating Highly Accurate Point Correspondences . . . . .	82
6.2.1	The Calibration Rig . . . . .	83
6.2.2	Spherical Calibration Target . . . . .	88
6.3	Imaging a Sphere . . . . .	92
6.4	Measurement Error of the Camera . . . . .	95
6.5	Localization of the Calibration Points in the Image Space . . . . .	99
6.6	Fitting Ellipse by Least Squares Analysis . . . . .	104

6.6.1	Algebraic Fit . . . . .	105
6.6.2	Geometric Fit . . . . .	105
6.7	Geometric Fit of Ellipse in Parametric Form . . . . .	106
6.8	Practicalities in Camera Calibration . . . . .	108
<b>7</b>	<b>Performance Evaluation of the Proposed Camera Calibration Technique</b>	<b>110</b>
7.1	Performance Evaluation by Monte-Carlo Simulation . . . . .	112
7.1.1	Synthesizing the Image of a Virtual Calibration Target . . . . .	113
7.1.2	Simulation Parameters . . . . .	115
7.1.3	Simulation Data Analysis . . . . .	116
7.1.4	Inferences from the Simulation Results . . . . .	137
7.2	Accuracy in Practical Camera Calibration . . . . .	140
7.2.1	Accuracy Evaluation . . . . .	140
<b>8</b>	<b>Surface Reconstruction</b>	<b>144</b>
8.1	Basic Principle of Structured Light Based Reconstruction . . . . .	146
8.2	Laser Calibration . . . . .	148
8.2.1	Mathematical Formulation . . . . .	151
8.3	Localizing the Laser Points . . . . .	152
8.3.1	Initial Approximation . . . . .	154
8.3.2	Data Reduction . . . . .	155
8.4	Experimental Surface Reconstruction . . . . .	156
<b>9</b>	<b>Conclusion</b>	<b>158</b>



<b>Bibliography</b>	<b>161</b>
<b>Appendices</b>	<b>169</b>
<b>A Newton-Gauss Algorithm for Non-Linear Least Squares Analysis</b>	<b>169</b>
<b>B Conversion Between Euler Angles and Quaternions</b>	<b>172</b>
B.1 Euler Angles to Quaternions . . . . .	173
B.2 Quaternions to Euler Angles . . . . .	173
<b>C Rotation Matrix to Quaternions Conversion</b>	<b>174</b>

# List of Tables

2.1	List of Camera Parameters . . . . .	22
3.1	Accuracy Evaluation of Camera Calibration Techniques: 3D Measure- ments (Armangué <i>et al.</i> ) . . . . .	30
6.1	Statistical Measures of the Distributions of Center Locations in the Image Space . . . . .	99
7.1	Camera Parameters for the Simulation (Analytic Camera) . . . . .	118
7.2	Lens Distortion Coefficients for the Analytic Lenses . . . . .	118
7.3	Simulation Parameters . . . . .	119
7.4	Simulation Results (Run $S_1W$ ) . . . . .	120
7.5	Simulation Results (Run $S_1S$ ) . . . . .	121
7.6	Simulation Results (Run $S_2W$ ) . . . . .	122
7.7	Simulation Results (Run $S_2S$ ) . . . . .	123
7.8	Simulation Results (Run $S_3W$ ) . . . . .	124
7.9	Simulation Results (Run $S_3S$ ) . . . . .	125
7.10	Comparison of the Estimates of the Intrinsic Camera Parameters Based on $APE$ . . . . .	126

7.11 Comparison of the Standard Errors in Pixels (Simulated Camera Calibration) . . . . .	138
7.12 Comparison of the Standard Errors in Pixels . . . . .	141
7.13 2D Measurements ( <i>pixels</i> ) . . . . .	143
8.1 Comparison of Sphericity Error . . . . .	157

# List of Figures

1.1	Architecture of the Surface Reconstruction System . . . . .	5
2.1	Pinhole Camera Model . . . . .	10
2.2	Field of View of a Camera and Captured Image . . . . .	11
2.3	Model of Image Formation in Camera . . . . .	13
2.4	Camera Model: As Seen from $Y_c$ Axis . . . . .	15
4.1	Radial Distortion: (a) Barrel distortion (b) Pin-cushion Distortion . .	48
4.2	Radial and Tangential Distortion . . . . .	49
5.1	Proposed Calibration Algorithm . . . . .	74
5.2	Camera Initialization GUI . . . . .	75
6.1	Typical Calibration Targets: Checkerboard Pattern (Left) and Dot Pattern (Right) . . . . .	78
6.2	Contemporary Calibration Targets . . . . .	80
6.3	Perspective Distortion in the Synthesized Image of a Square with Strong Lens Distortion Effect . . . . .	84



6.4	Perspective Distortion in the Synthesized Image of a Circle with Strong Lens Distortion Effect . . . . .	84
6.5	Calibration System . . . . .	85
6.6	GUI for Calibration Data Acquisition . . . . .	85
6.7	Calibration Rig . . . . .	86
6.8	Calibration Table . . . . .	87
6.9	Radiometric Properties of a Spherical Calibration Target and a Circular Calibration Target . . . . .	90
6.10	Spherical Calibration Target Mounted on the Calibration Table (Sphere Diameter = 25.4 mm) . . . . .	90
6.11	Perspective Projection of a Sphere (3D Model) . . . . .	91
6.12	Perspective Projection of a Sphere (Cross-section through a General Plane Containing the Center of the Sphere and the center of projection) . . . . .	92
6.13	Distribution of the Geometric Centers of the Images of the Spheres (Ceramic & Delrin) . . . . .	97
6.14	Histogram of $X$ and $Y$ Coordinates of the Geometric Centers of the Images of the Spheres (Ceramic & Delrin) . . . . .	98
6.15	Image Feature Extraction Technique . . . . .	100
6.16	3D Plot of the Gray Scale Image (Gray Values are Plotted in 3D Against Their Respective Pixel Positions) . . . . .	101
6.17	Histogram of a Generic Gray Scale Image of the Calibration Target . . . . .	101
6.18	Gradient Image with the Peaks Localized . . . . .	102
6.19	Gray Scale Intensity and the Corresponding First Derivative of the Spatial Signal in the Radial Direction . . . . .	102

6.20 Segmented Image (Localized Contour, Minimum Bounding Rectangle, Major Axis of Inertia) . . . . .	103
6.21 Fitting Ellipse to the Edge Points . . . . .	108
6.22 Superimposed Fitted Ellipse on Actual Calibration Image . . . . .	109
7.1 Virtual Calibration Target with $40 \times 40 \times 3$ Calibration Points . . . .	115
7.2 Defined Bound for Random Error . . . . .	116
7.3 Image of the Virtual Calibration Target . . . . .	117
7.4 <i>APE</i> of the Estimates of the Scale Factor $s$ ( $L_W$ ) . . . . .	127
7.5 <i>APE</i> of the Estimates of the Scale Factor $s$ ( $L_S$ ) . . . . .	127
7.6 <i>APE</i> of the Estimates of the Effective Focal Length $f$ ( $L_W$ ) . . . . .	127
7.7 <i>APE</i> of the Estimates of the Effective Focal Length $f$ ( $L_S$ ) . . . . .	128
7.8 <i>APE</i> of the Estimates of the Image Center Coordinate $u_0$ ( $L_W$ ) . . . .	128
7.9 <i>APE</i> of the Estimates of the Image Center Coordinate $u_0$ ( $L_S$ ) . . . .	128
7.10 <i>APE</i> of the Estimates of the Image Center Coordinate $v_0$ ( $L_W$ ) . . . .	129
7.11 <i>APE</i> of the Estimates of the Image Center Coordinate $v_0$ ( $L_S$ ) . . . .	129
7.12 <i>APE</i> of the Estimates of the Radial Lens Distortion Coefficient $k_1$ ( $L_W$ )	129
7.13 <i>APE</i> of the Estimates of the Radial Lens Distortion Coefficient $k_1$ ( $L_S$ )	130
7.14 <i>APE</i> of the Estimates of the Radial Lens Distortion Coefficient $k_2$ ( $L_W$ )	130
7.15 <i>APE</i> of the Estimates of the Radial Lens Distortion Coefficient $k_2$ ( $L_S$ )	130
7.16 <i>APE</i> of the Estimates of the Radial Lens Distortion Coefficient $p_1$ ( $L_W$ )	131
7.17 <i>APE</i> of the Estimates of the Radial Lens Distortion Coefficient $p_1$ ( $L_S$ )	131
7.18 <i>APE</i> of the Estimates of the Radial Lens Distortion Coefficient $p_2$ ( $L_W$ )	131
7.19 <i>APE</i> of the Estimates of the Radial Lens Distortion Coefficient $p_2$ ( $L_S$ )	132

7.20	<i>APE</i> of the Estimates of the Quaternion Component $d$ ( $L_W$ ) . . . . .	132
7.21	<i>APE</i> of the Estimates of the Quaternion Component $d$ ( $L_S$ ) . . . . .	132
7.22	<i>APE</i> of the Estimates of the Quaternion Component $a$ ( $L_W$ ) . . . . .	133
7.23	<i>APE</i> of the Estimates of the Quaternion Component $a$ ( $L_S$ ) . . . . .	133
7.24	<i>APE</i> of the Estimates of the Quaternion Component $b$ ( $L_W$ ) . . . . .	133
7.25	<i>APE</i> of the Estimates of the Quaternion Component $b$ ( $L_S$ ) . . . . .	134
7.26	<i>APE</i> of the Estimates of the Quaternion Component $c$ ( $L_W$ ) . . . . .	134
7.27	<i>APE</i> of the Estimates of the Quaternion Component $c$ ( $L_S$ ) . . . . .	134
7.28	<i>APE</i> of the Estimates of the Extrinsic Position Vector Component $t_x$ ( $L_W$ ) . . . . .	135
7.29	<i>APE</i> of the Estimates of the Extrinsic Position Vector Component $t_x$ ( $L_S$ ) . . . . .	135
7.30	<i>APE</i> of the Estimates of the Extrinsic Position Vector Component $t_y$ ( $L_W$ ) . . . . .	135
7.31	<i>APE</i> of the Estimates of the Extrinsic Position Vector Component $t_y$ ( $L_S$ ) . . . . .	136
7.32	<i>APE</i> of the Estimates of the Extrinsic Position Vector Component $t_z$ ( $L_W$ ) . . . . .	136
7.33	<i>APE</i> of the Estimates of the Extrinsic Position Vector Component $t_z$ ( $L_S$ ) . . . . .	136
7.34	Average Standard Error in Pixels ( $X$ Axis) . . . . .	137
7.35	Average Standard Error in Pixels ( $Y$ Axis) . . . . .	137
7.36	Residuals of the Calibrated Camera Models (Proposed & Heikkila) . .	142



8.1	Range Ambiguity in Perspective Projection . . . . .	145
8.2	Surface Reconstruction System Equipped with Laser Projector . . . .	148
8.3	Calibration Target for Laser Projector (Step Height 0.25 <i>inch</i> ) . . . .	149
8.4	Fitting Gaussian Curves to Data Points . . . . .	154
8.5	Reconstruction of A Spherical Surface (Sphere Diameter = 25.4 <i>mm</i> )	156



# Chapter 1

## Introduction

Since the wheel was invented in the 5th millennium BC, technology has influenced and changed human life. There is one thing, however, that remains constant for modern and ancient man. Despite all the differences in their physiques, intellects and everyday life challenges, both depend on vision as the primary sensory system. Though the gift of vision is taken for granted, a closer look reveals how important a role it plays in helping us to function effectively. First of all, besides auditory system, vision is the only other biologically available remote sensing system in the human anatomy. Moreover, it enables us to gather a large volume of data with a high acquisition rate. Considering all the advantages and potential of vision as a means of sensing, the idea of incorporating silicon based vision systems into machines occurred to the inventors since systematic research began in the field of artificial intelligence. As a result, *Machine Vision* emerged as a promising field in engineering. It can be defined as the systematic study and development of technologies that primarily aim to extract useful information from the images of a real world scene. This particular field

of engineering owes its birth and development to mathematics, computer engineering and digital signal processing.

Machine vision tasks include, but are not limited to, robot navigation, medical diagnosis, remote sensing, document processing and product inspection in manufacturing plants. These tasks are usually a general composition of some basic problems that are solved by machine vision techniques. 3D model acquisition or surface reconstruction is one such basic problem that retrieves the depth or the range of an object from image input. This particular subfield of vision studies originated from photogrammetry. Since the problem was addressed by the vision community, an extensive amount of work has been done and some impressive successes were achieved. Despite the invention and the development of a number of techniques for surface reconstruction, the landmark of attaining high accuracy is yet to be accomplished. In this work this challenge was undertaken with an aspiration to find a highly accurate, albeit cost effective, solution.

## 1.1 Problem Definition

The main objective of this work is to develop a vision based surface reconstruction system that features the following characteristics:

- The system may only receive images of the object as the input.
- It may employ additional means of non-contact nature to gather data that complements the image input (e.g., illuminating the scene with structured light).

This will assist in solving the basic range ambiguity problem associated with

monocular vision.

- The system has to be highly accurate. Attained accuracy should convincingly exhibit significant improvement over the existing systems.
- It should be cost efficient. Off-the-shelf digital cameras and lenses offer greater feasibility in this regard.
- The system is not required to perform in real-time since it will be a prototype. However, the design should be flexible enough so that it can be easily modified to operate in real-time conditions.

## 1.2 Challenges

Conventional digital cameras capture a  $2D$  interpretation of the  $3D$  world which is called the image. Once the  $3D$  world is mapped onto the  $2D$  image plane, the so called depth or range data is no longer available from the image. The surface reconstruction system attempts to retrieve the range by triangulation with the input from camera besides the laser projector. This is the basic operational principle of the system; however, in practice there are some challenges that need to be overcome first in order to achieve good performance:

- In order to recover a  $3D$  model of the object (i.e., reconstruct the surface of the object), the camera and the laser projector have to be calibrated *a priori*.
- An automated and accurate physical system is needed to produce and to collect the data for camera calibration, laser calibration and range finding. This system



should also be equipped with necessary control systems to actively control the camera and the laser projector.

- Consistent with the design requirement, the surface reconstruction system receives image input only. Consequently, data mining from the image input is another important aspect of the system. Customized image processing techniques (i.e., edge detection, blob analysis, fitting geometric primitives etc.) need to be developed and implemented to extract the useful data from the image input.

### 1.3 System Architecture

To build a system that meets the requirements stated in the preceding section, an off-the-shelf, industrial grade digital camera was chosen as the primary sensor that acquires image of the object whose 3D model is to be determined. To complement the image input, a laser projector that emits a laser light of linear profile was employed to retrieve the range of the object by triangulation. In addition, a desktop computer was commissioned to execute the following tasks:

- Generating the point correspondences for the camera calibration.
- Acquiring, processing and storing the calibration data.
- Controlling the calibration of the laser projector.
- Analyzing the data and subsequently building the 3D model.
- Providing interface between the user and the system.

Two auxiliary subsystems; namely, the calibration rig and the illumination source, provided additional functionalities to the system. The calibration rig is a physical structure that rigidly houses all the components of the system. It positions the object in the field of view of the camera accurately with two degrees of freedom in order to generate the point correspondences. The illumination source is comprised of four LED arrays that emit white light. It ensured uniform and controlled illumination over the field of view of the camera. Figure 1.1 illustrates the architecture of the surface reconstruction system.

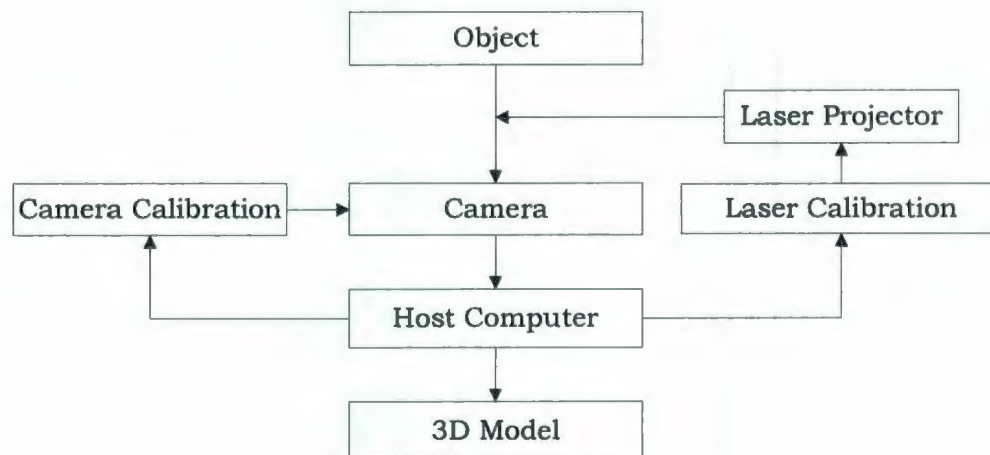


Figure 1.1: Architecture of the Surface Reconstruction System

## 1.4 Organization of thesis

This thesis is presented in nine chapters:

- **Chapter 1:** This chapter introduces the thesis. It defines the core problem



that the thesis attempts and lists the challenges associated with this problem.

The architecture of the surface reconstruction system is also discussed.

- **Chapter 2:** A geometrical interpretation of the image formation process in digital cameras (i.e., camera model) is presented in this chapter. The effect of lens distortion is modeled in order to develop a comprehensive mathematical representation of the imaging process.
- **Chapter 3:** The conventional camera calibration techniques are reviewed in Chapter 3. It also mentions reported comparative experimental studies assessing the accuracy of the calibration techniques.
- **Chapter 4:** A set of propositions are made to improve the conventionally adopted methodologies for camera calibration in this chapter. The propositions primarily include orientation representation and error minimization in the calibration problem.
- **Chapter 5:** A novel camera calibration technique is developed in Chapter 5. All the mathematical formulations of the solution to the calibration problem are documented.
- **Chapter 6:** This chapter deals with the implementation aspects of the proposed calibration technique. The mathematical analyses of various data mining techniques along with a brief description of the hardware are presented.
- **Chapter 7:** A thorough performance evaluation of the proposed camera calibration technique is conducted in Chapter 7. A comparative study on accuracy using computer simulation and as well as practical data is presented.

- **Chapter 8:** A novel and relatively simple laser calibration technique is developed. Following discussions include a significantly improved method of localizing the laser line in the image space. An object is digitized and the acquired 3D model is compared against the geometrical properties of the object determined by conventional metrological means.
- **Chapter 9:** Finally, Chapter 9 includes the concluding remarks and some propositions for future development.

## Chapter 2

# Camera Model: Mapping of 3D to 2D

In order to model the image formation process in cameras; i.e., perspective projection of a 3D real world scene onto a 2D image plane, geometric and optical characteristics of the camera and the scene are required to be known *a priori*. Geometric characteristics include knowledge of the position and orientation of the camera relative to the world coordinate system, image center and aspect ratio of the discretized image plane. Optical characteristics include focal length and distortion characteristics of the lens. Experimental determination of these characteristic parameters (henceforth referred to as camera parameters) is called camera calibration. These parameters constitute an analytical model of the image formation process that is often referred to as the camera model. In the following section a nomenclature is presented that is followed throughout the thesis.

## 2.1 Nomenclature

Before proceeding any further it is necessary to adopt a consistent nomenclature to avoid confusion. There are some ambiguities regarding usage of some camera model terms appearing in the literature.

- **Forward Camera Model:** Mapping of  $3D$  world coordinates onto  $2D$  image coordinates through perspective projection and lens distortion. This model provides the corresponding  $2D$  image coordinates for given  $3D$  world coordinates and camera parameters.
- **Backward Camera Model:** Intuitively, this model should provide the  $3D$  world coordinates from  $2D$  image coordinates and camera parameters; however, without the depth information, it is not possible to solve for the world coordinates explicitly. Therefore, this model doesn't have a unique solution.
- **Forward Distortion Model:** Adding distortion to the image coordinates given by pure forward perspective projection model (i.e., pinhole camera model).
- **Backward Distortion Model:** Provides undistorted (i.e., corrected) image coordinates; i.e., pure perspective projection, from distorted image coordinates.

## 2.2 Pinhole Camera Model

Machine vision text books and scholarly publications (e.g., [1, 2, 3, 4, 5, 6]) often use the Pinhole Camera Model (Figure 2.1) to develop a geometric interpretation of the image formation process. This mapping of  $3D$  world coordinates to  $2D$  image coor-



ordinates is quite simple. The light ray from the point in 3-Space passing through the camera projection center is projected onto the image plane. Intersection of the light ray and the image plane provides the image point. For example, in Figure 2.3 point  $P$  in 3-Space is projected as  $P'$ . Though this modeling approach ignores the effects of lens thickness and focus of the lens, reasonable approximation is still obtainable [2]. In ([3, 4, 5, 6]) the pinhole camera model was refined to account for image distortion due to lens geometry and inaccuracies in lens assembly.

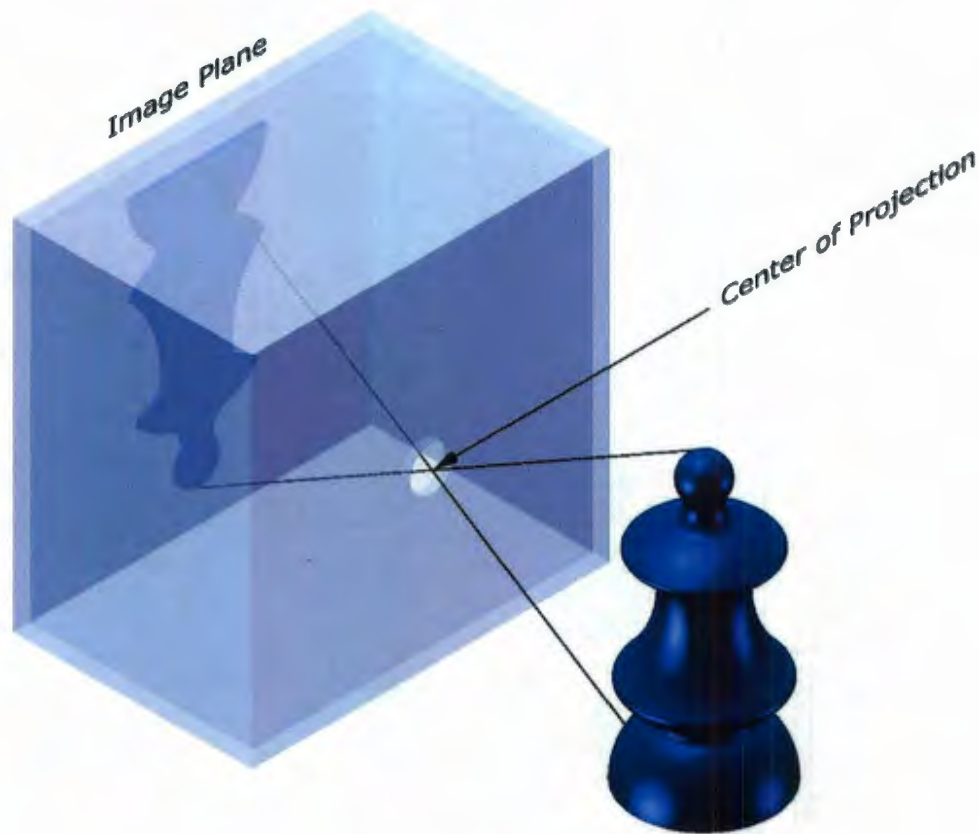


Figure 2.1: Pinhole Camera Model



## 2.3 Modeling Image Formation

A typical digital image, when stored, is conventionally defined with respect to a 2D coordinate frame  $UV$ , the so-called *image coordinate frame* (Figure 2.2). When the image is viewed on a screen, axes  $U$  and  $V$  are the horizontal and the vertical axes respectively. Other coordinate frames in the image formation model will be defined with a reference to the image coordinate frame.

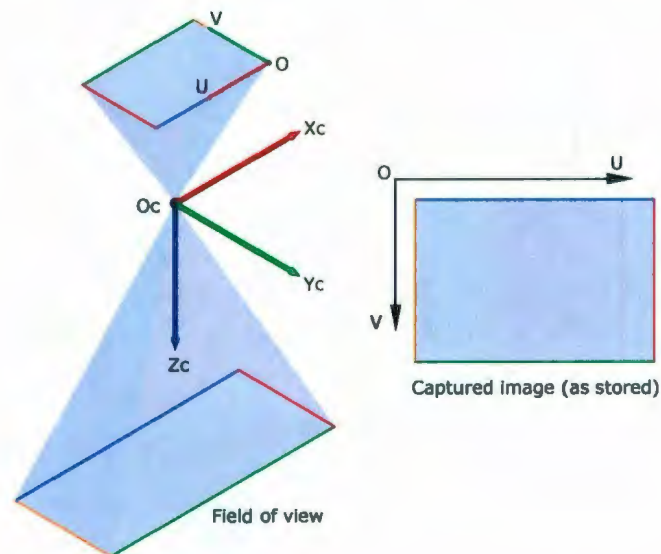


Figure 2.2: Field of View of a Camera and Captured Image

The camera model for this work was set up in the following manner (Figure 2.3):

- **Image Coordinate Frame:** This 2D coordinate frame has its origin  $O$  at the left upper corner of the image when stored or viewed. This definition agrees with the conventional image coordinate frame in computers.
- **Retinal Coordinate Frame:** This is a 2D coordinate frame with  $X_i Y_i$  axes. The

origin of this coordinate frame is at the image center  $O_i$ , the intersection point of the optical axis of the lens and the image plane. This coordinate frame is defined so that  $X_i$  and  $Y_i$  axes are respectively in the direction of axes  $U$  and  $V$ .

- **Camera Coordinate Frame:** This coordinate frame  $X_c Y_c Z_c$  has its origin at the principal point or the center of projection of the lens  $O_c$ . The  $X_c Y_c$  plane is parallel to the image plane and the  $Z_c$  axis coincides with the principal axis (or the aiming vector) of the lens. Axes  $X_c$  and  $Y_c$  are in the opposite direction of retinal axes  $X_i$  and  $Y_i$  respectively. The distance from the principal point  $O_c$  to the image center  $O_i$  is the effective focal length  $f$  of the camera. It should be mentioned that this convention was adopted to more accurately represent the actual camera geometry that inverts the image.
- **World Coordinate Frame:** This coordinate frame  $X_w Y_w Z_w$  defines the 3D object position and orientation in world coordinates. This coordinate frame is unconstrained.

## 2.4 Forward Camera Model

The basic goal of the forward camera model is to find the distorted image coordinates from the 3D world coordinates for a given set of camera parameters. This model maps 3D world coordinates onto the 2D image plane for a given camera. This mapping can be decomposed into two Euclidean and one perspective projective transformation and one nonlinear transformation for lens distortion. These transformations take place in

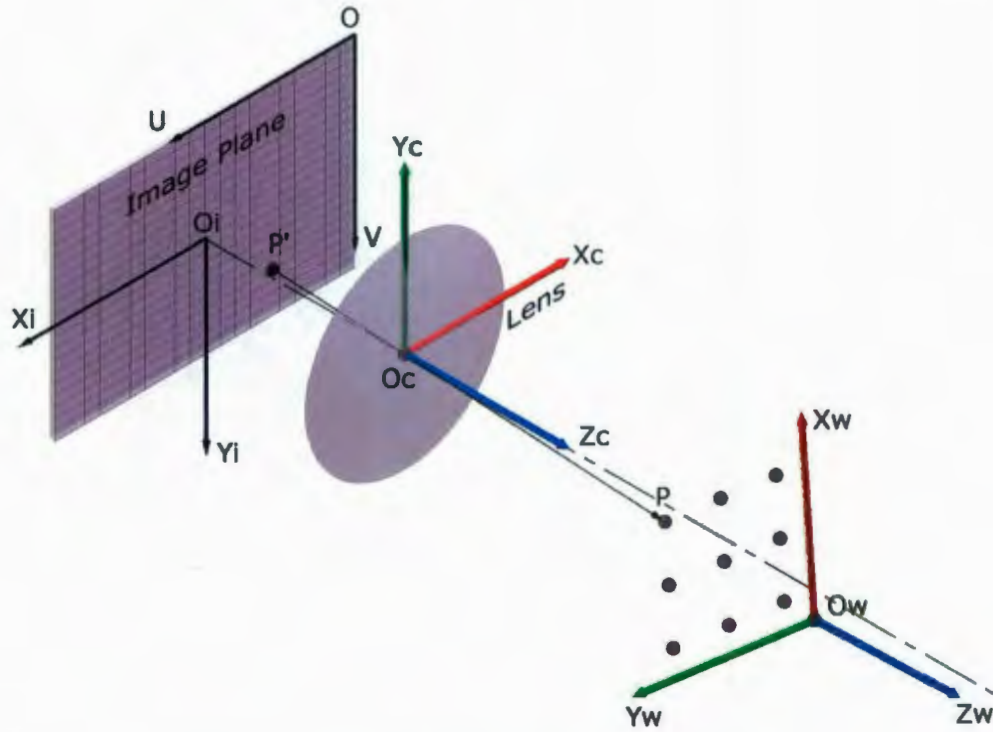


Figure 2.3: Model of Image Formation in Camera

the order explained in the following sections.

#### 2.4.1 Transformation from World Coordinate Frame to Camera Coordinate Frame

Let the coordinates of point  $P$  in Figure 2.3 be  $(x_w, y_w, z_w)$  measured with respect to the world coordinate frame  $X_w Y_w Z_w$ ; i.e., position vector of point  $P$  in world coordinate frame is  $[x_w \ y_w \ z_w]^T$ . The goal is to transform this position vector to the 3D camera coordinate frame  $X_c Y_c Z_c$ . To facilitate the computation of this transformation the homogeneous coordinates  ${}^wP = [x_w \ y_w \ z_w \ 1]^T$  is used. This



is particularly advantageous as one matrix multiplication can take care of both the rotation and the translation parts of the transformation. This can be achieved by using Equation 2.1.

$$\begin{aligned} {}^cP &= {}^cT_w \times {}^wP \\ &= \begin{bmatrix} {}^cR_w & {}^c t_w \\ 0 & 1 \end{bmatrix} \times {}^wP \end{aligned} \quad (2.1)$$

Here,

${}^cT_w$  =  $4 \times 4$  transformation matrix from the world coordinate frame to the camera coordinate frame

${}^cR_w$  =  $3 \times 3$  rotation matrix from the world coordinate frame to the camera coordinate frame

${}^c t_w$  = Position vector of the origin  $O_w$  of the world coordinate frame in the camera coordinate frame

## 2.4.2 Perspective Projection Transformation from Camera Coordinate Frame to Retinal Coordinate Frame

In this step, we try to obtain the  $2D$  representation of the  $3D$  world. This  $2D$  representation is called the image of the  $3D$  world. Figure 2.4 clarifies the process of projection. Note that this figure is identical to the model presented in Figure 2.3, only viewed from  $Y_c$  axis.

From Figure 2.4, we see that  $\triangle PO_cQ \sim \triangle P'O_cO_i$ . This similarity provides an opportunity to establish a geometric relationship between the coordinates of object point  $P$  ( $x_c, y_c, z_c$ ) in the camera coordinate frame  $X_cY_cZ_c$  and the coordinates of



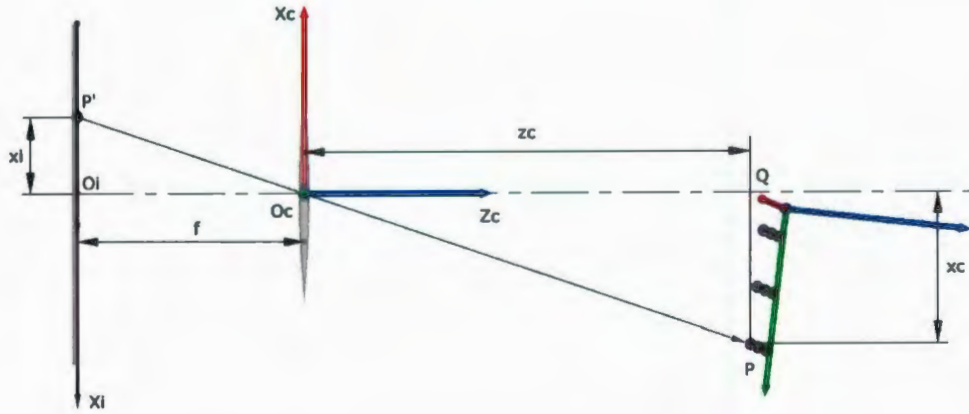


Figure 2.4: Camera Model: As Seen from  $Y_c$  Axis

the image point  $P'$  ( $x_i, y_i$ ) in the retinal coordinate frame  $X_i Y_i$ . From equality of the ratios of similar sides of similar triangles, we obtain the following relationship (Equation 2.2):

$$\begin{aligned} \frac{x_i}{f} &= \frac{x_c}{z_c} \\ \Rightarrow x_i &= \frac{x_c}{z_c} f \end{aligned} \quad (2.2)$$

Equation 2.2 is true when the pixel spacings in the horizontal and the vertical directions in the image sensor are equal which might not be the general case. To account for this, a scale factor  $s$  will be introduced which is the ratio of pixel spacing in the horizontal and the vertical directions. A more general relation is Equation 2.3 which includes the scale factor  $s$ :

$$\begin{aligned} \frac{x_i}{f} &= s \frac{x_c}{z_c} \\ \Rightarrow x_i &= s \frac{x_c}{z_c} f \end{aligned} \quad (2.3)$$

Using similar reasoning and viewing the model from the negative direction of the axis

$X_c$  provides the following relationship (Equation 2.4):

$$\begin{aligned}\frac{y_i}{f} &= \frac{y_c}{z_c} \\ \Rightarrow y_i &= \frac{y_c}{z_c} f\end{aligned}\tag{2.4}$$

Equations 2.3 and 2.4 yield the projective transformation.

### 2.4.3 Transformation from Retinal Coordinate Frame to Image Coordinate Frame

Since the axes of the retinal and image coordinate frames are in the same direction, transformation from the  $X_i Y_i$  coordinate frame to the  $UV$  coordinate frame is pure translation. The coordinates of the origin of the  $X_i Y_i$  coordinate frame are already defined as the image center  $O_i (u_0, v_0)$  in the coordinate frame  $UV$ . Therefore the general transformation can be formulated as in Equation 2.5.

$$\begin{bmatrix} u \\ v \\ 1 \end{bmatrix} = \begin{bmatrix} 1 & 0 & u_0 \\ 0 & 1 & v_0 \\ 0 & 0 & 1 \end{bmatrix} \times \begin{bmatrix} x_i \\ y_i \\ 1 \end{bmatrix}\tag{2.5}$$

$$= \begin{bmatrix} x_i + u_0 \\ y_i + v_0 \\ 1 \end{bmatrix}$$

Henceforth the coordinates  $(u, v)$  will be referred to as the undistorted image coordinates. Now the camera matrix [2] will be formulated. This is also referred to as the perspective projection matrix [4]. Substituting Equations 2.3 and 2.4 in

Equation 2.5 yields Equation 2.6.

$$\begin{aligned}
 \begin{bmatrix} u \\ v \\ 1 \end{bmatrix} &= \begin{bmatrix} \frac{x_c}{z_c} f s + u_0 \\ \frac{y_c}{z_c} f + v_0 \\ 1 \end{bmatrix} \\
 &= \begin{bmatrix} x_c f s + u_0 z_c \\ y_c f + v_0 z_c \\ z_c \end{bmatrix} \\
 &= \begin{bmatrix} s f & 0 & u_0 & 0 \\ 0 & f & v_0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix} \times \begin{bmatrix} x_c \\ y_c \\ z_c \\ 1 \end{bmatrix} \\
 &= C \times {}^c P
 \end{aligned} \tag{2.6}$$

The camera matrix  $C$  in Equation 2.6 performs the projective transformation and translation of the camera coordinates. It should be noted that, the product of the matrix-vector multiplication in Equation 2.6 yields the homogeneous coordinates of the corresponding image point. In order to obtain the equivalent Cartesian coordinates of the image point the third component of the homogeneous coordinates must be normalized to unity.

#### 2.4.4 From Undistorted Image Coordinates to Distorted Image Coordinates

Lens distortion results from imperfect lens construction and inaccurate lens assembly. Distortion causes the theoretical image point provided by the pinhole camera model

to be shifted on the image plane either tangentially or radially or both. To model the image formation process accurately, one has to compensate for this phenomenon. A brief survey of lens distortion modeling is given in Chapter 4. Various mathematical models of lens distortion have been proposed in the literature (e.g., [7, 8]). In this work the distortion model given in [7] has been adopted since this lens distortion model is better suited for conventional lenses. Let the undistorted image coordinates of point  $P'$  be  $a_u = [u \ v]^T$  and the corresponding distorted image coordinates be  $a_d = [u_d \ v_d]^T$ . The relationship between distorted and undistorted image coordinates is given by Equation 2.7:

$$\begin{aligned}
 a_u &= a_d + \mathcal{F}(a_d, \delta) \\
 &= \begin{bmatrix} u_d \\ v_d \end{bmatrix} + \begin{bmatrix} \bar{u}_d(k_1 r_d^2 + k_2 r_d^4 + k_3 r_d^6 + \dots) + (2p_1 \bar{u}_d \bar{v}_d + p_2(r_d^2 + 2\bar{u}_d^2))(1 + p_3 r_d^2 + \dots) \\ \bar{v}_d(k_1 r_d^2 + k_2 r_d^4 + k_3 r_d^6 + \dots) + (p_1(r_d^2 + 2\bar{v}_d^2) + 2p_2 \bar{u}_d \bar{v}_d)(1 + p_3 r_d^2 + \dots) \end{bmatrix} \\
 \text{Here, } \bar{u}_d &= u_d - u_0, \quad \bar{v}_d = v_d - v_0, \quad r_d = \sqrt{\bar{u}_d^2 + \bar{v}_d^2} \\
 \text{and } \delta &= [k_1, k_2, \dots, p_1, p_2, \dots]^T
 \end{aligned} \tag{2.7}$$

The parameters  $k_1, k_2, \dots$  and  $p_1, p_2, \dots$  are the coefficients of radial and tangential distortion respectively. Since the higher order coefficients of both radial and tangential distortion are very small, they can be neglected without significant loss of accuracy [9]; however, for a wide angle lens higher order terms might be significant. In the Equation 2.8 the simplified lens distortion model is presented:

$$\begin{bmatrix} u \\ v \end{bmatrix} = \begin{bmatrix} u_d \\ v_d \end{bmatrix} + \begin{bmatrix} \bar{u}_d(k_1 r_d^2 + k_2 r_d^4) + (2p_1 \bar{u}_d \bar{v}_d + p_2(r_d^2 + 2\bar{u}_d^2)) \\ \bar{v}_d(k_1 r_d^2 + k_2 r_d^4) + (p_1(r_d^2 + 2\bar{v}_d^2) + 2p_2 \bar{u}_d \bar{v}_d) \end{bmatrix} \tag{2.8}$$



As discussed in Section 2.1, the distortion model presented in Equation 2.8 is referred to as the backward distortion model. Note that a closed-form analytic solution for the distorted image coordinates in terms of the undistorted image coordinates (i.e., the forward distortion model) is not readily obtainable from Equation 2.8. The forward distortion model, however, must be incorporated into the forward camera model for estimating lens distortion effects. To solve this problem, a numerical technique is employed to determine the distorted image coordinates from their undistorted counterparts; i.e., to solve Equation 2.8 for  $[u_d \ v_d]$  given  $[u \ v]$ . Equation 2.8 provides a system of two nonlinear equations which must be solved for the unknown parameter vector  $\bar{\theta} = [u_d \ v_d]$  (Equation 2.9):

$$\begin{aligned} F_u &= \bar{u}_d(k_1 r_d^2 + k_2 r_d^4) + (2p_1 \bar{u}_d \bar{v}_d + p_2(r_d^2 + 2\bar{u}_d^2)) - u \\ F_v &= \bar{v}_d(k_1 r_d^2 + k_2 r_d^4) + (p_1(r_d^2 + 2\bar{v}_d^2) + 2p_2 \bar{u}_d \bar{v}_d) - v \end{aligned} \quad (2.9)$$

In order to obtain a numerical solution to Equation 2.9 through application of the Newton-Gauss method, the Jacobian matrix for the system of equations needs to be first calculated. The Jacobian matrix is given by (Equation 2.10):

$$J = \begin{bmatrix} \frac{\partial}{\partial u} F_u & \frac{\partial}{\partial v} F_u \\ \frac{\partial}{\partial u} F_v & \frac{\partial}{\partial v} F_v \end{bmatrix} \quad (2.10)$$

The elements of the Jacobian matrix are expanded in Equation 2.11.

$$J_{11} = 1 + k_1(\bar{u}^2 + \bar{v}^2) + k_2(\bar{u}^2 + \bar{v}^2)^2 + \bar{u}^2(k_1 + 2k_2(\bar{u}^2 + \bar{v}^2)) + 2p_1\bar{v} + 6p_2\bar{u}$$

$$J_{12} = 2\bar{u}\bar{v}(k_1 + 2k_2(\bar{u}^2 + \bar{v}^2)) + 2p_1\bar{u} + 2p_2\bar{v}$$

$$J_{21} = 2\bar{u}\bar{v}(k_1 + 2k_2(\bar{u}^2 + \bar{v}^2)) + 2p_1\bar{u} + 2p_2\bar{v}$$

$$J_{22} = 1 + k_1(\bar{u}^2 + \bar{v}^2) + k_2(\bar{u}^2 + \bar{v}^2)^2 + 2\bar{v}^2(k_1 + 2k_2(\bar{u}^2 + \bar{v}^2)) + 6p_1\bar{v} + 2p_2\bar{u}$$

Here,  $\bar{u} = u_d - u_0$  and  $\bar{v} = v_d - v_0$

(2.11)

The distorted image coordinates  $[u_d \ v_d]^T$  are determined in an iterative fashion from the nonlinear system in Equation 2.9. Based on the initial parameter vector  $\theta_0 = [u \ v]^T$ , the algorithm converges to a solution in a few iterations.

### 2.4.5 Overall Transformation

The transformations explained in the previous sections can now be combined to form the forward camera model. This will be achieved in two steps. In the first step, the world coordinates are transformed into undistorted image coordinates. Substituting Equation 2.1 into Equation 2.6 yields the first step of the desired transformation.

$$\begin{bmatrix} u \\ v \\ 1 \end{bmatrix} = C \times {}^cT_w \times {}^wP \quad (2.12)$$

Subsequently, the distorted image coordinates are estimated from the undistorted image coordinates using the forward distortion model; i.e., through the application

of the Newton-Gauss method.

$$\begin{bmatrix} u_d \\ v_d \end{bmatrix} \xleftarrow{\text{Nonlinear Optimization}} \begin{bmatrix} u \\ v \end{bmatrix} \quad (2.13)$$

## 2.5 Camera Parameters

In the course of developing the forward camera model, a set of camera parameters have been defined that characterize the camera geometrically and optically. These parameters are summarized in Table 2.1.

Since parameter 1 and parameter 2 in Table 2.1 describe the external geometry of the camera, they are called the extrinsic camera parameters. The remaining camera parameters characterize the internal geometry and optics of the camera; therefore, they are referred to as the intrinsic camera parameters. It should also be noted that specifying the orientation of a  $3D$  coordinate frame with respect to another  $3D$  coordinate frame requires three independent parameters. Similarly the position vector is also composed of three independent parameters; hence, there are six independent extrinsic camera parameters.

Table 2.1: List of Camera Parameters

Parameter	Definition
1 ${}^cR_w$	Rotation matrix defining the orientation of the world coordinate frame
2 ${}^ct_w$	Position vector of the origin of the world coordinate frame in the camera coordinate frame
3 $f$	Focal length of the camera
4 $s$	Scale factor
5 $[u_0 \ v_0]$	Position vector of the origin of the retinal coordinate frame in the image coordinate frame
6 $[k_1 \ k_2]$	Coefficients of radial distortion of the lens
7 $[p_1 \ p_2]$	Coefficients of tangential distortion of the lens



## Chapter 3

# Camera Calibration Methods: A Review

Accuracy and performance of vision based 3D measurement systems largely depend upon the quality of the camera calibration since the camera is the primary and sometimes the only sensor in such systems. Over the years in photogrammetry and computer/machine vision literature an extensive body of work on camera models and camera calibration techniques has been proposed and reported. Excellent reviews of the developments in camera calibration in the past years can be found in [10, 11, 12]. The history of developing camera models and calibration techniques closely follows the history of camera and imaging technology development itself. In the early years of photogrammetry, film based cameras were used and aerial photogrammetry was the primary field of application, where accuracy was of secondary importance. Moreover, relatively low resolution of the cameras also limited the performance. Eventually *close range photogrammetry* became a prominent research topic that attempted to

solve the problem of acquiring metric information from the image of a real world scene. By then camera resolution was high enough for the lens distortion to be significant and researchers started to augment camera models with lens distortion terms. Camera calibration methods in computer/machine vision have close ties with calibration methods in photogrammetry. Precursory work in close range photogrammetry, [13] inspired many of the widely adopted camera models proposed more recently (e.g., [3, 4, 5, 6]). It also introduced one of the most popular and widely used calibration methods at that time which is known as “self calibrating bundle adjustment” [11]. Low cost, off-the-shelf Charged Couple Device (CCD) based digital cameras became widely available for machine vision applications with the advancement of the imaging technologies. One of the implicit assumptions of the self calibrating method is that the centre of the optical axis is very close to the centre of the imaging sensor; however, that is not necessarily valid for CCD cameras [12]. Hence the geometry of modern CCD cameras was no longer conducive to self calibration. As a result there was a renewed research interest for developing stand alone photogrammetric calibration approaches, especially fully automated calibration [11]. In the following sections a brief overview of existing models and various calibration techniques is highlighted. The camera models reported in the literature generally use a perspective projection model; therefore, only the differences in the camera models will be emphasized.

### **3.1 Classification of Camera Calibration Methods**

Camera calibration methods and their governing models as reported in the literature to date can be classified according to several different criteria. A brief account of

camera calibration methods, models and their classification can be found in [10, 11]. The subsequent discussions will concentrate on [11] since it provides an excellent comprehensive classification of the camera models developed to date.

Almost all reported camera models are based on the fundamental principles of projective geometry. Reference [11] classifies the camera models based on the modelling approach. The two classes are:

- Camera model based on perspective projection: These models follow the perspective projection model which is very similar to the model described in Chapter 2. The inherent implication in such models is that the internal orientation of the image plane is stable for a given focal length. These models also accommodate perturbations from collinearity, both linear and non-linear, mostly due to lens distortion. Calibration techniques following these models require five or more point correspondences within a multi image network for nonlinear least squares bundle adjustment; however, better estimation of the camera parameters is expected when a large number of point correspondence metric data is available. An optimum number of points can be experimentally determined beyond which the quality of estimation does not significantly increase. Determining the minimum number of correspondences that is required for solving the camera model is interesting from a theoretical point of view, but from a practical point of view the minimum number of point correspondences does not provide an optimal solution.
- Camera model based on Essential & Fundamental Matrices: These models are more focused on projective geometry than Euclidean scene reconstruction. They



are characterized by the essential and fundamental matrix models and can accommodate variable and unknown focal length. Nonlinearities arising from lens distortion are difficult to deal with in these models [11].

Both [11] and [10] categorize the camera models into two classes, *implicit* and *explicit* camera models, based on whether or not the process of image formation is modeled physically. *Implicit* camera models try to quantify the mapping of the 3D world coordinates to the 2D image coordinates by the multiplication of a column vector by a single matrix. The vector is a homogeneous representation of 3D coordinates and the matrix is a  $3 \times 4$  matrix that is often referred to as the *camera matrix*. Elements of the camera matrix do not correspond to any physical parameters related to camera geometry, orientation or optics. Examples of such models are presented in [14] and [15]. On the other hand, *explicit* camera models try to develop the model defined by physical parameters of the camera. Examples are [3, 4, 5, 13].

Further classification of camera models is possible by choosing different criteria. Whether the calibration method uses 3D rather than planar point arrays can also be a basis of classification. While [4] can accommodate both 3D and planar point arrays, [5] and [16] uses only planar point arrays. Another possible criterion for classification can be parameter estimation and optimization techniques employed in the camera calibration method. According to this criterion for classification, camera models and calibration techniques can be divided into three major classes as discussed in the following.

- Underlying models of calibration methods using linear techniques neglect the nonlinear lens distortion. These methods can be fast and can also include



closed-form solutions. They simplify the camera model a great deal but at the cost of accuracy [11]. One example of this class is the well known Direct Linear Transformation (DLT) method [17].

- Calibration methods using nonlinear optimization techniques attempt to find a rigorous and accurate representation of the camera internal orientation through iterative least squares estimation. The governing camera model includes lens distortion. Reference [13] is typical of such a calibration method.
- Some calibration methods use a combination of linear and nonlinear techniques. Linear methods are employed to estimate the initial approximations of the camera parameters. These parameters are then used as an initial parameter vector for a nonlinear search to iteratively refine the estimation. Linear search for initial approximation increases the probability of convergence of the nonlinear search. Examples are described in [3, 6, 18, 19].

## 3.2 Camera Calibration Techniques in Computer Vision

Previous sections dealt with camera models and calibration techniques in general. The scope of the subsequent discussion will be limited to calibration methods and models which are prominent and widely used in computer/machine vision. A comparative experimental performance evaluation was carried out in [20] for camera calibration techniques presented in [3, 5, 17]. A similar study can be found in [10] on camera calibration techniques presented in [3, 6, 15, 18]; however, the most recent review [11],

recognizes [3, 5, 19] as the most commonly adopted calibration techniques in Computer Vision. All three techniques are based on a pinhole camera model and include terms for radial distortion. They also employ a nonlinear optimization technique for parameter estimation. Traditionally these methods employ a so-called calibration target; e.g., checkerboard pattern or an array of circular feature points (also referred to as control points in [4]) arranged in either a 3D or a coplanar fashion.

Tsai developed a two step calibration technique ([3]) for CCD based digital cameras. This technique assumes that some of the camera parameters will be provided by the manufacturer to reduce the computational cost associated with the initial approximation. Euler angles were employed to represent orientation. Tsai's technique also attempted to compensate for the timing error associated with image scanning and acquisition in the camera system. He also assumed that no matter what kind of distortion is present in the lens sensor assembly, the total effects of lens distortion are compensated for as radial distortion. One or more views of 3D or coplanar calibration targets of known world coordinates is used in this technique. A minimum of eight feature points per image are required to solve the calibration problem with a set of linear equations based on the *Radial Alignment Constraint* (RAC).

Heikkila & Silven developed another two step calibration technique described in [19]. Using DLT, the closed form solution gives initial estimates of the camera parameters. By applying the Levenberg-Marquardt algorithm this technique then estimates the first order gradients (i.e., the Jacobian matrix) and the intrinsic parameters and lens distortion are refined. This technique, which employs two coefficients for radial distortion and two coefficients for tangential or decentering distortion, works with either a single 3D calibration target or multiple coplanar calibration grids. Heikkila

improved this calibration technique in [4] by implementing implicit correction for the projective distortion of the circular control points.

Zhang's calibration technique [5] uses a 2D checkerboard pattern. Images of this calibration target are acquired for at least two different orientations and used for calibration. The vertices of the checkerboard pattern are used as control points. The relative coordinates of the control points are assumed to be known *a priori* and the algorithm computes the projective transformation of these points to within a scale factor. In this method, third and fourth order terms of radial distortion terms are recovered. Refinement of the recovered parameters is achieved through nonlinear minimization of the reprojection error solved using Levenberg-Marquardt algorithm. Reference [11] remarks that this method is very similar to [16].

### 3.3 Experimental Evaluation of Calibration Techniques

Many attempts to study the performance of different calibration methods experimentally have been reported in the literature. For example, various works such as [10, 20, 21, 22] can be cited. In this section the conclusions of these experiments are summarized. Studies which specifically deal with calibration techniques for machine vision will be highlighted.

In [10] the accuracy of both linear and nonlinear calibration techniques was evaluated by Armangué *et al.* based on 3D and 2D measurements. Reference [15] was chosen as an example of a typical linear technique while nonlinear examples were



drawn from [3, 6, 18]. For 3D measurements two evaluation criteria; namely, distance with respect to the optical ray and Normalized Stereo Calibration Error (NSCE), were used. It was found that in terms of these two accuracy criteria, performance of nonlinear methods were superior to the linear ones. From the experimental data presented in Table 3.1 the authors concluded that, though nonlinear methods are computationally expensive and time consuming, improved accuracy justifies their use.

Table 3.1: Accuracy Evaluation of Camera Calibration Techniques: 3D  
Measurements (Armangué *et al.*)

	Ref.	3D Measurements (mm)			NSCE
		Mean	$\sigma$	Max	
Hall	[15]	0.1615	0.1028	0.5634	$n/a$
Faugeras	[18]	0.1811	0.1357	0.8707	0.6555
Tsai	[3, 23]	0.0564	0.0305	0.1626	0.2033
Weng	[6]	0.0570	0.0305	0.1696	0.2064

A study that dealt with the presence of noise in calibration data and the corresponding effects on calibration quality was conducted in [21] using two popular techniques presented in [3, 5]. Simulation and real-life experiments facilitated this study. Normalized Calibration Error (NCE) [6] was used as the evaluation criterion. The main conclusions in [21] were:

- Performance of [3] degrades as the noise level in the calibration data increases;



however, with low noise levels the calibration quality is better than [5]. Achieving a low noise level can be a rigorous process.

- In terms of the complexity of implementation versus performance, [5] is a good choice since data acquisition is simple and the effect of noise level can be negated by increasing the number of calibration points. Since this is a method based on a planar calibration targets, producing these targets is comparatively easy.
- Including the higher order terms in the lens distortion model (e.g., fourth or sixth order) can potentially degrade calibration quality especially when the noise level in the data is comparatively high. Higher order terms only provide better accuracy when the noise level is low.

To assist in the task of localizing the calibration points in the image, the calibration target was illuminated by structured light in the study conducted in [22]. The study was conducted using Direct Linear Transform (DLT) as well as the two nonlinear methods presented in [3] and [5]. The conclusions presented in [22] are consistent with those presented in [10, 21].

## Chapter 4

# Propositions for Improvement in Methodology

In machine vision, generic camera calibration techniques attempt to solve the underlying camera model from the point correspondences between the object space and the image space. These techniques provide estimates of both the orientation and position of the calibration target (i.e., *extrinsic parameters*) as well as the optical and geometric properties of the camera defining the perspective projection (i.e., *intrinsic parameters*). Solving the camera model from the point correspondences is approached numerically in the general calibration problem. The following discussion proposes some improvements on the existing methodologies for achieving better performance in terms of convergence and accuracy.

## 4.1 Representation of Orientation

Euler's theorem states that the most general displacement of a rigid body with a fixed point in space is equivalent to a single rotation of the body about an axis through the fixed point [24]. This displacement is usually termed as *pure rotation* in the literature (e.g., [25]). At least three independent parameters are required to represent pure rotation; e.g., *Euler Angles*. In references ([3, 19, 4]) Euler Angles were used to quantify relative orientation of one coordinate frame with respect to another coordinate frame in Euclidean 3-Space. A rotation matrix was formulated from the Euler angles. In reference [26], it was shown that no 3 dimensional parameterizations can be both global and non-singular; i.e., every rotation determines some finite values of the parameters but these values are not uniquely defined [26]. This phenomenon results in *gimbal lock singularities* where two of the three Euler angles belong to the same Degree of Freedom (DOF) with the loss of one DOF [27]. This is the major drawback of using Euler angles for parameterizing rotations; however, Euler angles are commonly employed to formulate the  $3 \times 3$  rotation matrix used to represent camera orientation. This introduces redundancy since any general rotation in 3-Space has only three degrees of freedom whereas the rotation matrix has nine elements of which only three are independent. The rotation matrix is a *special orthogonal matrix* which has a determinant of +1 and with rows and columns that are orthonormal [28]. Let a generic element of the rotation matrix be represented by  $R_{ij}$  where this element belongs to the  $i^{\text{th}}$  row and the  $j^{\text{th}}$  column with the following constraint on  $i$  and  $j$ :

$$1 \leq i, j \leq 3$$



Since the rotation matrix is orthogonal, the product of the matrix by its transpose must yield the identity matrix. This additional constraint results in six independent equations that can be used to enforce orthogonality on the rotation matrix (Equation 4.1).

$$\begin{aligned}
R_{11}^2 + R_{12}^2 + R_{13}^2 &= 1 \\
R_{21}^2 + R_{22}^2 + R_{23}^2 &= 1 \\
R_{31}^2 + R_{32}^2 + R_{33}^2 &= 1 \\
R_{11}R_{12} + R_{21}R_{22} + R_{31}R_{32} &= 0 \\
R_{11}R_{13} + R_{21}R_{23} + R_{31}R_{33} &= 0 \\
R_{12}R_{13} + R_{22}R_{23} + R_{32}R_{33} &= 0
\end{aligned} \tag{4.1}$$

In a nonlinear optimization problem, explicitly maintaining these six orthogonality constraints (Equation 4.1) is very difficult [29]; hence, it would be desirable to find an alternative non-redundant and non-singular parametrization which is not possible according to reference [26]. Therefore the most suitable parametrization is the one where there is no singularity over the space of 3-Rotations and where the redundancy is kept to a minimum [29]. One such parametrization is the *unit quaternions*. In 1992 quaternions were used in the camera calibration technique proposed by [6] to represent camera orientation. In [6] quaternions were used only for linear optimization of the calibration problem in order to obtain an initial approximation of the camera parameters for the subsequent nonlinear optimization; whereas this work utilizes quaternions for nonlinear optimization. Besides not being singular over the space of rotations, the quaternion representation is more computationally efficient than the traditional Euler angle representation. Unlike Euler angles, rotations represented by



quaternions do not involve any computationally expensive transcendental functions. A brief mathematical account on the unit quaternions and quaternion algebra is given in the following section.

## 4.2 Unit Quaternions: Non-singular Representation of Rotation

Firstly the vector space of quaternions will be defined. Let the set  $A = \{1, \hat{i}, \hat{j}, \hat{k}\}$  be a basis of the vector space  $V$  of dimension four over the set of real numbers  $\mathbb{R}$ . The elements in the basis set are respectively the scalar unit 1 and three mutually perpendicular unit vectors  $\hat{i}, \hat{j}$  and  $\hat{k}$ . These elements obey the laws of combination as formulated in Equation 4.2.

$$\begin{aligned}\hat{i}^2 &= \hat{j}^2 = \hat{k}^2 = -1 \\ \hat{i}\hat{j} &= \hat{k}, \quad \hat{j}\hat{i} = -\hat{k} \\ \hat{j}\hat{k} &= \hat{i}, \quad \hat{k}\hat{j} = -\hat{i} \\ \hat{k}\hat{i} &= \hat{j}, \quad \hat{i}\hat{k} = -\hat{j}\end{aligned}\tag{4.2}$$

The quaternion  $q$  is a linear combination of the elements in the basis set  $A$  (Equation 4.3).

$$q = d + a\hat{i} + b\hat{j} + c\hat{k}, \quad a, b, c, d \in \mathbb{R}\tag{4.3}$$

### 4.2.1 Quaternion Algebra

In this section basic quaternion algebra will be reviewed. Let two quaternions be given by  $q_1 = d_1 + a_1\hat{i} + b_1\hat{j} + c_1\hat{k}$  and  $q_2 = d_2 + a_2\hat{i} + b_2\hat{j} + c_2\hat{k}$ . The basic definitions

of quaternion algebra are listed below:

- Equality of two quaternions (i.e.,  $q_1 = q_2$ ) is satisfied if and only if:

$$d_1 = d_2, a_1 = a_2, b_1 = b_2, \text{ and } c_1 = c_2$$

- The sum of two quaternions  $q_1$  and  $q_2$  is defined as:

$$q_1 + q_2 = (d_1 + d_2) + (a_1 + a_2)\hat{i} + (b_1 + b_2)\hat{j} + (c_1 + c_2)\hat{k}$$

- The difference of two quaternions is defined as:

$$q_1 - q_2 = (d_1 - d_2) + (a_1 - a_2)\hat{i} + (b_1 - b_2)\hat{j} + (c_1 - c_2)\hat{k}$$

- When a quaternion  $q_1$  is multiplied by a scalar  $\lambda$ , the product  $q_\lambda$  is defined as:

$$q_\lambda = \lambda d_1 + \lambda a_1 \hat{i} + \lambda b_1 \hat{j} + \lambda c_1 \hat{k}$$

- Finally, the quaternion  $q_1$  is a zero quaternion, if and only if:

$$d_1 = 0, a_1 = 0, b_1 = 0, \text{ and } c_1 = 0$$

The product of two quaternions,  $q_1$  and  $q_2$ , can be developed using these basic definitions and the laws of combination of the unit vectors (Equation 4.2):

$$q_1 q_2 = (d_1 + a_1 \hat{i} + b_1 \hat{j} + c_1 \hat{k})(d_2 + a_2 \hat{i} + b_2 \hat{j} + c_2 \hat{k})$$

Applying the distributive law yields:

$$q_1 q_2 = d_1 d_2 + a_1 a_2 \hat{i}^2 + b_1 b_2 \hat{j}^2 + c_1 c_2 \hat{k}^2 + d_1 a_2 \hat{i} + d_1 b_2 \hat{j} + d_1 c_2 \hat{k} + a_1 d_2 \hat{i} + a_1 b_2 \hat{j} \\ + a_1 c_2 \hat{i} \hat{k} + b_1 d_2 \hat{j} + b_1 a_2 \hat{j} \hat{i} + b_1 c_2 \hat{j} \hat{k} + c_1 d_2 \hat{k} + c_1 a_2 \hat{k} \hat{i} + c_1 b_2 \hat{k} \hat{j}$$

Applying the definitions from Equation 4.2, and subsequently collecting common terms yields:

$$q_1 q_2 = d_1 d_2 - a_1 a_2 - b_1 b_2 - c_1 c_2 + d_1 (a_2 \hat{i} + b_2 \hat{j} + c_2 \hat{k}) + d_2 (a_1 \hat{i} + b_1 \hat{j} + c_1 \hat{k}) \\ + \begin{vmatrix} \hat{i} & \hat{j} & \hat{k} \\ a_1 & b_1 & c_1 \\ a_2 & b_2 & c_2 \end{vmatrix} \quad (4.4)$$

Let  $q = q_1 q_2$ , where,  $q = d + a\hat{i} + b\hat{j} + c\hat{k}$ . Equating the coefficients in the different components of the quaternion from Equation 4.4 yields Equation 4.5.

$$\begin{aligned} d &= d_1 d_2 - a_1 a_2 - b_1 b_2 - c_1 c_2 \\ a &= d_1 a_2 + d_2 a_1 + b_1 c_2 - b_2 c_1 \\ b &= d_1 b_2 + d_2 b_1 + c_1 a_2 - c_2 a_1 \\ c &= d_1 c_2 + d_2 c_1 + a_1 b_2 - a_2 b_1 \end{aligned} \quad (4.5)$$

In general quaternion multiplication is not commutative; i.e.,  $q_1 q_2 \neq q_2 q_1$ . The exception occurs only when the final determinant in Equation 4.4 vanishes.

### 4.2.2 Conjugate, Norm, Inverse

The conjugate of a quaternion  $q = d + a\hat{i} + b\hat{j} + c\hat{k}$ , which is denoted by  $q'$ , is defined as follows (Equation 4.6):

$$q' = d - a\hat{i} - b\hat{j} - c\hat{k} \quad (4.6)$$

The product of a quaternion  $q$ , and its conjugate  $q'$ , is commutative and yields a scalar. This product is shown in Equation 4.7.

$$qq' = q'q = d^2 + a^2 + b^2 + c^2 \quad (4.7)$$

The product of a quaternion and its conjugate in Equation 4.7 is referred to as the norm of the quaternion  $q$ , and is denoted by  $N_q$ . The quaternion  $q$  is referred to as a *unit quaternion* if the norm of  $q$  is unity; i.e.,  $N_q = 1$ . The definition of the norm of a quaternion is formulated in Equation 4.8.

$$N_q = qq' = q'q \quad (4.8)$$

The reciprocal of a quaternion is defined as its inverse. For any unit quaternion (i.e.,  $N_q = 1$ ), it can be shown from Equation 4.8 that the conjugate of a unit quaternion is also its inverse.

$$q^{-1} = q' \quad (4.9)$$

### 4.2.3 Rotation Representation

Quaternions have been used in robotics and vision for rigid body pose estimation (e.g., [30, 31]). Let us consider a pure rotation of a rigid body about an axis  $\bar{n} = n_x\hat{i} + n_y\hat{j} + n_z\hat{k}$ , and through an angle  $\phi$ . Four parameters can be defined to represent this rotation (Equation 4.10).

$$a = n_x \sin \frac{\phi}{2}, \quad b = n_y \sin \frac{\phi}{2}, \quad c = n_z \sin \frac{\phi}{2}, \quad d = \cos \frac{\phi}{2} \quad (4.10)$$

These four parameters can be expressed in quaternion notation; i.e.,  $q = d + a\hat{i} + b\hat{j} + c\hat{k}$ . This quaternion  $q$  represents the rotation about the axis  $\bar{n}$  by an angle  $\phi$ . Next it will



be shown that the quaternion  $q$  is a unit quaternion; i.e., the norm of the quaternion  $N_q$  is unity (Equation 4.11).

$$\begin{aligned}
 N_q &= d^2 + a^2 + b^2 + c^2 = \cos^2 \frac{\phi}{2} + n_x^2 \sin^2 \frac{\phi}{2} + n_y^2 \sin^2 \frac{\phi}{2} + n_z^2 \sin^2 \frac{\phi}{2} \\
 &= \cos^2 \frac{\phi}{2} + \sin^2 \frac{\phi}{2} (n_x^2 + n_y^2 + n_z^2) \\
 &= \cos^2 \frac{\phi}{2} + \sin^2 \frac{\phi}{2} \quad [\because \bar{n} \text{ is a unit vector, } \|\bar{n}\| = 1] \\
 &= 1
 \end{aligned} \tag{4.11}$$

Let a vector  $\bar{r} = r_x \hat{i} + r_y \hat{j} + r_z \hat{k}$  be transformed through pure rotation about the axis  $\bar{n}$  by an angle  $\phi$  resulting in the transformed vector  $\bar{r}'$ . Using quaternion algebra a general formulation for this transformation will be derived in the following discussion. It will be shown that the transformation of a vector by a pure rotation represented by the unit quaternion  $q$  can be accomplished by the quaternion rotation operator; i.e.,  $q(\ )q^{-1}$ , (Equation 4.12).

$$\bar{r}' = q\bar{r}q^{-1} \tag{4.12}$$

In the subsequent analysis the vector  $\bar{r}$  can be regarded as a quaternion whose scalar part is zero. The quaternion multiplication in Equation 4.12 will be performed in two steps. Firstly, let the partial product quaternion be  $t = q\bar{r}$ . From the general formula for quaternion multiplication (Equation 4.4),  $t$  can be expanded as follows:

$$\begin{aligned}
 t = q\bar{r} &= -\sin \frac{\phi}{2} (n_x r_x + n_y r_y + n_z r_z) + \cos \frac{\phi}{2} (r_x \hat{i} + r_y \hat{j} + r_z \hat{k}) \\
 &\quad + \sin \frac{\phi}{2} \begin{vmatrix} \hat{i} & \hat{j} & \hat{k} \\ n_x & n_y & n_z \\ r_x & r_y & r_z \end{vmatrix}
 \end{aligned} \tag{4.13}$$

If the quaternion  $t$  is expanded as  $t = t_d + t_a\hat{i} + t_b\hat{j} + t_c\hat{k}$ , equating the coefficients from Equation 4.13 yields the following:

$$\begin{aligned} t_d &= -\sin \frac{\phi}{2} (n_x r_x + n_y r_y + n_z r_z) \\ t_a &= r_x \cos \frac{\phi}{2} + \sin \frac{\phi}{2} (n_y r_z - n_z r_y) \\ t_b &= r_y \cos \frac{\phi}{2} + \sin \frac{\phi}{2} (n_z r_x - n_x r_z) \\ t_c &= r_z \cos \frac{\phi}{2} + \sin \frac{\phi}{2} (n_x r_y - n_y r_x) \end{aligned} \quad (4.14)$$

The inverse of the quaternion  $q$  equals its conjugate since it is a unit quaternion. Therefore,

$$q^{-1} = q' = \cos \frac{\phi}{2} - \sin \frac{\phi}{2} (n_x\hat{i} + n_y\hat{j} + n_z\hat{k})$$

Now substituting the Equations 4.13 and 4.14 into Equation 4.12 yields:

$$\begin{aligned} \bar{r}' &= q\bar{r}q^{-1} \\ &= tq^{-1} \\ &= t_d \cos \frac{\phi}{2} + \sin \frac{\phi}{2} (t_a n_x + t_b n_y + t_c n_z) + t_d \sin \frac{\phi}{2} (-n_x\hat{i} - n_y\hat{j} - n_z\hat{k}) \\ &\quad + \cos \frac{\phi}{2} (t_a\hat{i} + t_b\hat{j} + t_c\hat{k}) + \sin \frac{\phi}{2} \begin{vmatrix} \hat{i} & \hat{j} & \hat{k} \\ t_a & t_b & t_c \\ -n_x & -n_y & -n_z \end{vmatrix} \end{aligned} \quad (4.15)$$

The scalar part of the transformed vector  $\bar{r}'$  can be written as:

$$S_{\bar{r}'} = t_d \cos \frac{\phi}{2} + \sin \frac{\phi}{2} (t_a n_x + t_b n_y + t_c n_z) \quad (4.16)$$

Substituting the expressions for  $t_a, t_b$  and  $t_c$  (from the Equation set 4.14) into Equation 4.16 yields:

$$S_{\bar{r}'} = -\sin \frac{\phi}{2} \cos \frac{\phi}{2} (n_x r_x + n_y r_y + n_z r_z) + \sin \frac{\phi}{2} [n_x (r_x \cos \frac{\phi}{2} + \sin \frac{\phi}{2} (n_y r_z - n_z r_y)) \\ + n_y (r_y \cos \frac{\phi}{2} + \sin \frac{\phi}{2} (n_z r_x - n_x r_z)) + n_z (r_z \cos \frac{\phi}{2} + \sin \frac{\phi}{2} (n_x r_y - n_y r_z))] \quad (4.17)$$

Expanding Equation 4.17 reduces it to  $S_{\bar{r}'} = 0$ . Since the scalar component  $S_{\bar{r}'}$  of the quaternion  $\bar{r}'$  is zero, the quaternion  $\bar{r}'$  is a vector. Thus the quaternion rotation operator  $q(\ )q^{-1}$  transforms a vector to another vector. The norms on the right and left sides of Equation 4.12 can be written as:

$$N_{\bar{r}'} = N_q N_{\bar{r}} N_{q^{-1}} \quad (4.18)$$

The order of the multiplication can be rearranged on the right side of Equation 4.18. Because of the fact that the quaternions  $q$ , and  $q^{-1}$  are both unit quaternions; i.e.,  $N_q = N_{q^{-1}} = 1$ , Equation 4.18 simplifies to:

$$N_{\bar{r}'} = N_{\bar{r}} \quad (4.19)$$

It is apparent that the norm of the initial vector  $\bar{r}$  is equal to that of the transformed vector  $\bar{r}'$ ; i.e., the magnitudes of the vectors are equal. Since this is only possible when the transformation is a pure rotation, it can be concluded that the operator  $q(\ )q^{-1}$  represents a pure rotation.

Now it will be shown in two steps that this rotation is indeed a pure rotation about the axis  $\bar{n} = n_x \hat{i} + n_y \hat{j} + n_z \hat{k}$ , by an angle  $\phi$ . First, a rotation matrix will be developed from Equation 4.15, which is just an alternative formulation of the transformation given by the quaternion rotation operator  $q(\ )q^{-1}$ . Secondly, this rotation matrix will

be compared to the rotation matrix provided by the axis angle representation of the same rotation. Equality of the two rotation matrices will prove that the quaternion rotation operator  $q(\ )q^{-1}$  accomplishes the desired transformation.

Substituting the expressions for  $t_d, t_a, t_b$  and  $t_c$  from Equation 4.14 into Equation 4.15, and subsequent regrouping and rearranging terms yield Equation 4.20. The following short hand notations are used in Equation 4.20.

$$c_t = \cos \frac{\phi}{2}, s_t = \sin \frac{\phi}{2}$$

$$\begin{aligned} \bar{r}' = & \hat{i}\{s_t^2[n_x(n_x r_x + 2n_y r_y + 2n_z r_z) + r_x(-n_y^2 - n_z^2)] + 2s_t c_t(n_y r_z - r_y n_z) + c_t^2 r_x\} \\ & + \hat{j}\{s_t^2[n_y(2n_x r_x + n_y r_y + 2n_z r_z) + r_y(-n_x^2 - n_z^2)] + 2s_t c_t(n_z r_x - n_x r_z) + c_t^2 r_y\} \\ & + \hat{k}\{s_t^2[n_z(2n_x r_x + 2n_y r_y + n_z r_z) + r_z(-n_x^2 - n_y^2)] + 2s_t c_t(n_x r_y - n_y r_x) + c_t^2 r_z\} \end{aligned} \quad (4.20)$$

Since the vector  $\bar{n}$  representing the axis of rotation is a unit vector, the expression,  $n_x^2 + n_y^2 + n_z^2 = 1$  holds true. From this expression the following relationships can be derived:

$$\begin{aligned} -n_y^2 - n_z^2 &= n_x^2 - 1 \\ -n_x^2 - n_z^2 &= n_y^2 - 1 \\ -n_x^2 - n_y^2 &= n_z^2 - 1 \end{aligned} \quad (4.21)$$

Substituting these expressions into Equation 4.20 yields:

$$\begin{aligned} \bar{r}' = & \hat{i}\{s_t^2[n_x(2n_x r_x + 2n_y r_y + 2n_z r_z) - r_x] + 2s_t c_t(n_y r_z - r_y n_z) + c_t^2 r_x\} \\ & + \hat{j}\{s_t^2[n_y(2n_x r_x + 2n_y r_y + 2n_z r_z) - r_y] + 2s_t c_t(n_z r_x - n_x r_z) + c_t^2 r_y\} \\ & + \hat{k}\{s_t^2[n_z(2n_x r_x + 2n_y r_y + 2n_z r_z) - r_z] + 2s_t c_t(n_x r_y - n_y r_x) + c_t^2 r_z\} \end{aligned} \quad (4.22)$$



Substituting Equation 4.10, and the unit quaternion norm condition; i.e.,  $d^2 + a^2 + b^2 + c^2 = 1$ , into the  $\hat{i}$  component of the vector  $\bar{r}'$  into Equation 4.22 provides Equation 4.23. The following trigonometric identity is used in Equation 4.23 to facilitate algebraic manipulation:

$$c_t^2 - s_t^2 = 2c_t^2 - 1$$

$$\begin{aligned}\bar{r}'_{\hat{i}} &= s_t^2[n_x(2n_xr_x + 2n_yr_y + 2n_zr_z) - r_x] + 2s_tc_t(n_yr_z - r_yr_z) + c_t^2r_x \\ &= n_xs_t(2n_xs_tr_x + 2n_ys_tr_y + 2n_zs_tr_z) + r_x(c_t^2 - s_t^2) + 2c_t(n_ys_tr_z - r_yr_z) \\ &= n_xs_t(2n_xs_tr_x + 2n_ys_tr_y + 2n_zs_tr_z) + r_x(2c_t^2 - 1) + 2c_t(n_ys_tr_z - r_yr_z) \quad (4.23) \\ &= a(2ar_x + 2br_y + 2cr_z) + r_x(2d^2 - (d^2 + a^2 + b^2 + c^2)) + 2d(br_z - cr_y) \\ &= a(2ar_x + 2br_y + 2cr_z) + r_x(d^2 - a^2 - b^2 - c^2) + 2d(br_z - cr_y) \\ &= (d^2 + a^2 - b^2 - c^2)r_x + (2ab - 2cd)r_y + (2ca + 2bd)r_z\end{aligned}$$

The  $\hat{j}$  and  $\hat{k}$  components of  $\bar{r}'$  can be expressed in linear combinations of  $r_x, r_y$  and  $r_z$  after executing similar algebraic manipulations and substitutions.

$$\begin{aligned}\bar{r}'_{\hat{j}} &= (2ab + 2cd)r_x + (d^2 - a^2 + b^2 - c^2)r_y + (2bc - 2ad)r_z \\ \bar{r}'_{\hat{k}} &= (2ca - 2bd)r_x + (2bc + 2ad)r_y + (d^2 - a^2 - b^2 + c^2)r_z\end{aligned} \quad (4.24)$$

Let us consider two coordinate frames; one being the space fixed coordinate frame  $A$  and the other being the body fixed coordinate frame  $B$ . A rotation matrix can now be formulated from Equations 4.23 and 4.24. Prior to the rotation both of the coordinate frames were aligned with coincident origins. The vector  $\bar{r} = {}^B\mathbf{r} = r_x\hat{i} + r_y\hat{j} + r_z\hat{k}$  is expressed with respect to the body fixed coordinate frame  $B$ . After rotating the vector  $\bar{r}$  about the axis  $\bar{n}$  by an angle  $\phi$ , the rotated vector  $\bar{r}' = {}^A\mathbf{r}$  is expressed with respect to the space fixed coordinate frame  $A$ . The transformation can now be formulated as the following:

$${}^A\mathbf{r} = {}^A R_B \times {}^B\mathbf{r}$$

The rotation matrix  ${}^A R_B$  in the above equation is:

$${}^A R_B = \begin{bmatrix} d^2 + a^2 - b^2 - c^2 & 2ab - 2cd & 2ca + 2bd \\ 2ab + 2cd & d^2 - a^2 + b^2 - c^2 & 2bc - 2ad \\ 2ca - 2bd & 2bc + 2ad & d^2 - a^2 - b^2 + c^2 \end{bmatrix} \quad (4.25)$$

Axis angle representation of the same rotation can be defined by the rotation matrix  ${}^A R'_B$  (Equation 4.26) [32]. The following short hand notations are used for convenience:

$$v = 1 - \cos \phi, \quad s' = \sin \phi \text{ and } c' = \cos \phi$$

$${}^A R'_B = \begin{bmatrix} n_x^2 v + c' & n_x n_y v - n_z s' & n_x n_z v + n_y s' \\ n_x n_y v + n_z s' & n_y^2 v + c' & n_y n_z v - n_x s' \\ n_x n_z v - n_y s' & n_y n_z v + n_x s' & n_z^2 v + c' \end{bmatrix} \quad (4.26)$$

To show that the matrices  ${}^A R'_B$  and  ${}^A R_B$  are equal, corresponding elements in  ${}^A R'_B$  and  ${}^A R_B$  are required to be equal. Simple trigonometric and algebraic manipulations can prove this equality. For example, the element in  ${}^A R'_B$ , belonging to the 1<sup>st</sup> row and the 1<sup>st</sup> column can be manipulated using half angle trigonometric identities and substituting these identities in the Equations 4.10 and 4.11, as follows:

$$\begin{aligned} n_x^2 v + c' &= n_x^2 (1 - \cos \phi) + \cos \phi \\ &= 2n_x^2 \sin^2 \frac{\phi}{2} + 2 \cos^2 \frac{\phi}{2} - 1 \\ &= 2a^2 + 2d^2 - (d^2 + a^2 + b^2 + c^2) \\ &= d^2 + a^2 - b^2 - c^2 \end{aligned}$$

Thus it was shown that the element in the 1<sup>st</sup> row and the 1<sup>st</sup> column of the matrix  ${}^A R'_B$ , equals the corresponding element of the matrix  ${}^A R_B$ . Similar algebraic manipulations and substitutions can show that each element in  ${}^A R'_B$  is equal to the corresponding element in  ${}^A R_B$ . This effectively proves that:

$${}^A R_B = {}^A R'_B$$

It can also be concluded that the quaternion rotation operator  $q(\ )q^{-1}$  provides the specified rotation given by the quaternion,  $q$ .

#### 4.2.4 In Context of the Camera Model

Equation 4.25 provides the rotation matrix which transforms position vectors in the body fixed coordinate frame to the space fixed coordinate frame given a general rotation. This transformation can be incorporated into the camera model (Chapter 2) for transforming position vectors (i.e., coordinates) of the points in the world coordinate frame to the camera coordinate frame. It is preferable to consider the camera coordinate frame  $C$  as the space fixed coordinate frame and world coordinate frame  $W$  as the body fixed coordinate frame. If the relative orientation of the world coordinate frame  $W$  with respect to the camera coordinate frame  $C$  is specified by the unit quaternion  $q = d + a\hat{i} + b\hat{j} + c\hat{k}$ , the general rotation matrix of the world coordinate frame  $W$  with respect to the camera coordinate frame  $C$  then becomes:

$${}^C R_W = \begin{bmatrix} d^2 + a^2 - b^2 - c^2 & 2ab - 2cd & 2ca + 2bd \\ 2ab + 2cd & d^2 - a^2 + b^2 - c^2 & 2bc - 2ad \\ 2ca - 2bd & 2bc + 2ad & d^2 - a^2 - b^2 + c^2 \end{bmatrix} \quad (4.27)$$



## 4.3 Modeling Lens Distortion

The failure of a lens element to produce a mathematically perfect image point is referred to as *Lens Aberration*. In the absence of chromatic lens aberrations (i.e., purely monochromatic light), lens aberrations can be broadly classified as follows [7]:

- Spherical Aberration
- Coma
- Astigmatism
- Curvature of field
- Distortion

In the context of a measurement system, distortion is the most important lens aberration since only distortion concerns imperfect image location rather than image formation. Distortion leads to a variation of the scale of an image as a function of position on the image plane which can result in significant measurement error when using a camera for metrology. Therefore, it is highly advisable to model lens distortion mathematically in critical measurement applications. While compensating for lens distortion within the camera model can substantially improve the accuracy of the measurement system, the nonlinear nature of such models adds to the computational cost and system complexity.

Lens distortion can be considered as a perturbation of the image coordinates from the linear pinhole camera model. For the sake of convenience in the subsequent discussion, the image coordinates provided by the pinhole camera model will be referred



to as the ray-traced image point. Based on this nomenclature lens distortion can be divided into two major categories:

- **Radial Distortion:** This type of distortion causes inward or outward displacement of the ray-traced image point. This is mainly caused by flawed curvature of the lens element. A negative radial distortion effect causes the outer image points to crowd to the image center. This phenomenon is referred to as *barrel distortion*. A positive radial distortion effect causes the inner points to spread outwardly and is referred to as *pin-cushion distortion*. In Figure 4.1, the square is the ray-traced image of a square grid in the object space where the image plane and the grid plane are parallel. Depending on the type of the distortion present in the optics of the camera, the real image can be either (a) or (b); i.e., barrel or pin-cushion distortion respectively. In barrel distortion the scale decreases, whereas, the scale increases in pin-cushion distortion. Radial distortion is strictly symmetrical about the optical axis [6].
- **Tangential/decentering Distortion:** Tangential or decentering distortion causes the ray-traced image points to be displaced tangentially; i.e., perpendicular to the radial lines. Noncollinear alignment of the optical centers of the lens elements in a complex lens system results in tangential distortion. It has been found that the tangential distortion of any line passing through the image centre is nearly symmetric in terms of magnitude and sign [7]. In Figure 4.2, this type of distortion effect is shown along with radial distortion.

All the prominent camera calibration techniques model lens distortion to some degree; e.g., ([4, 3, 5, 6]). References [3] and [5] modeled only radial lens distortion;

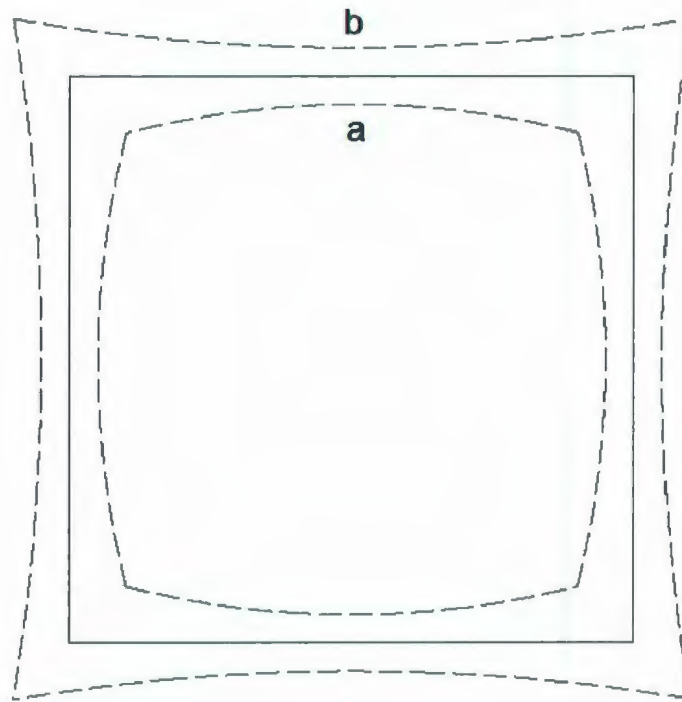


Figure 4.1: Radial Distortion: (a) Barrel distortion (b) Pin-cushion Distortion

whereas, a comprehensive lens distortion model that includes three types of lens distortion (radial, tangential and thin prism distortion) was used in [6]. The most recent calibration technique developed by Heikkila in [4], models both tangential and radial distortion. Although it is preferable to model lens distortion as accurately as possible, an exhaustive model can render the numerical search for the optimal camera parameters to be unnecessarily complicated. Lens distortion modeling thus becomes a tradeoff between accuracy and complexity.

The distortion modeling approach in [4] is described in Section 2.4.4. Heikkila in [4] included two radial and two tangential distortion coefficients. Higher order terms can be neglected without any significant loss of accuracy since they are only

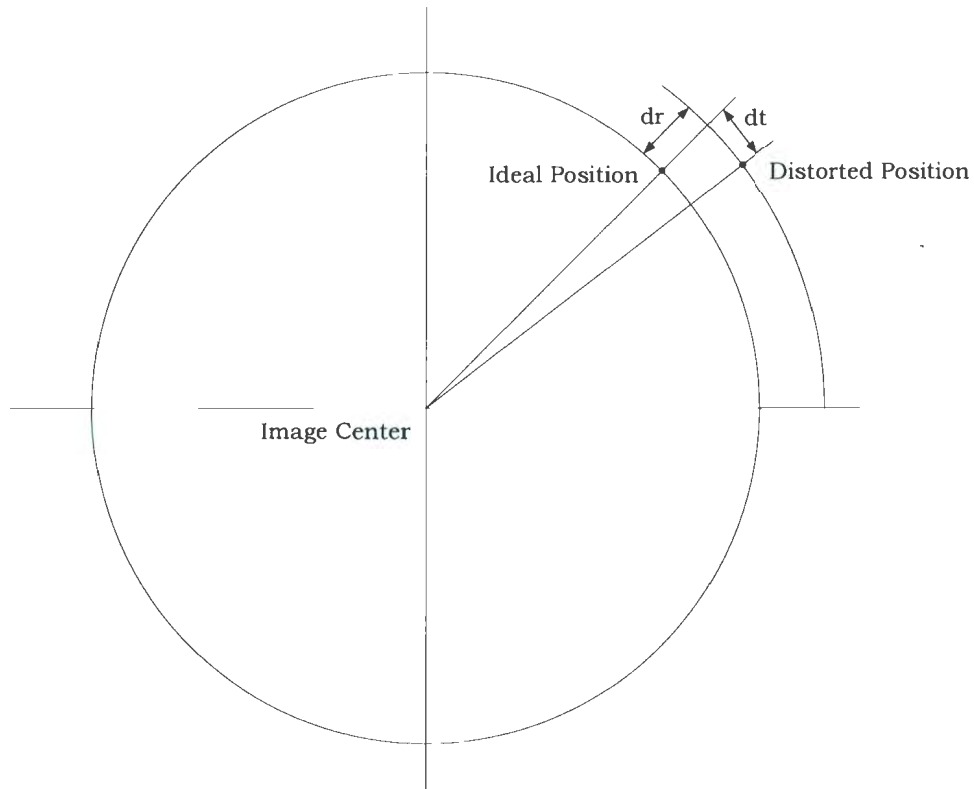


Figure 4.2: Radial and Tangential Distortion

significant in the case of wide angled lenses. The lens distortion model in Equation 2.7 provides the undistorted (i.e., ray-traced image coordinates) from the distorted image coordinates. It is often necessary, however, to find an expression for the distorted image coordinates in terms of the ideal image coordinates. The transformation from distorted to ideal image coordinates is referred to as the reverse distortion model. It is evident from Equation 2.7 that a closed-form solution for the inverse distortion model is not readily available. In [33] the following recursive approximation of Equation 2.7

was proposed in an attempt to address this problem:

$$\begin{aligned}
a_d &\approx a_u - \mathcal{F}(a_d, \delta) \\
&\approx a_u - \mathcal{F}(a_u - \mathcal{F}(a_d, \delta), \delta) \\
&\approx a_u - \mathcal{F}(a_u - \mathcal{F}(a_d - \mathcal{F}(a_d, \delta), \delta), \delta) \\
&\approx \dots
\end{aligned} \tag{4.28}$$

This recursive approximation is, however, computation intensive and hence impractical. Heikkila attempted to resolve this problem by expanding Equation 2.7 using a first-order Taylor series expansion (Equation 4.29).

$$a_u \approx a_d + \mathcal{F}(a_d, \delta) + \mathbf{D}(a_u)(a_d - a_u) \tag{4.29}$$

Heikkila further simplified this approximation by neglecting  $\mathbf{D}(a_u)$ , since  $\mathbf{D}(a_u) \ll 1$ .

A novel solution to this problem was proposed in Section 2.4.4 where the Newton-Gauss method was employed to numerically find the distorted image coordinates from the undistorted image coordinates without approximation. A *Computer Algebra System (CAS)* was used to find the exact analytic expression of the Jacobian matrix in the numerical iterative search. CAS packages such as Maple<sup>TM</sup> and Mathematica<sup>®</sup> are very useful software tools for symbolic algebraic calculations in such applications. For details, refer to Section 2.4.4.

## 4.4 Numerical Solution to the Calibration Problem

Depending on whether or not the calibration technique includes terms for lens distortion the camera calibration technique can be either linear or nonlinear in nature.



While linear methods are fast, less computation intensive and without convergence issues, nonlinear methods yield a more accurate model of the image formation process at the cost of solving a complex nonlinear system. Traditional methods [4, 5] often resort to the Levenberg-Marquardt algorithm [34]. Reference [35] investigated the efficiency and suitability of this algorithm when applied to machine vision problems. Though [35] does not offer any concrete conclusion, the Levenberg-Marquardt algorithm is found to be time consuming. Moreover, this algorithm does not compute the Jacobian of the nonlinear system of equations; rather it approximates it to enhance convergence. However, in context of a measurement system, convergence is of secondary importance while the primary goal is to estimate the camera parameters as accurately as possible. In the light of this design philosophy, it is well justified to employ the analytically derived Jacobian matrix in the numerical solution rather than approximating it. Admittedly the large number of unknowns and complex nature of the equations render the task virtually impossible to accomplish through manual algebraic manipulations. The application of CAS systems to derive the Jacobian analytically is a feasible alternative.

## 4.5 Choice of the Error Criterion

When the camera model is solved numerically, an error term must be minimized iteratively. This error can be geometric (i.e., the distance between the coordinates of the actual image points and the image points provided by the camera model [3, 4]) or algebraic (i.e., the difference between two algebraic expressions). The algebraic error term usually does not have any physical meaning [5]. When geometric distance

is considered, Euclidean Reconstruction is emphasized; therefore, minimizing the geometric error is justified when calibrating vision based measurement systems. This geometric distance can be interpreted either as the absolute distance between the actual image points and the projected image points or as two separate terms representing the coordinate components. Reference [29] postulates that the nonlinear iterative numerical techniques perform best when they have access to the individual error terms resulting in better accuracy in the camera parameters. In addition, the geometric distance can be minimized in either the distorted or the undistorted image space. Since the reverse lens distortion model does not have a closed-form solution, minimizing the error in the undistorted space is preferable and is the approach that will be adopted in this thesis.

## Chapter 5

# An Accurate Camera Calibration Technique

In this chapter a novel camera calibration technique is proposed that realizes the propositions made in Chapter 4. This calibration technique attempts to solve the general calibration problem numerically given  $n$  point correspondences between the object space and the distorted image space.

### 5.1 Formulation of the Calibration Problem

It is necessary to formulate the calibration problem mathematically to develop the framework for the numerical solution. In Equation 2.12 the perspective projection of a point in 3-Space  $[x_w, y_w, z_w]^T$  to the undistorted image point  $[u, v]^T$  was established under the pinhole camera model. Equation 2.12 is expanded and separated into the individual  $u, v$  coordinate components in Equations 5.1 and 5.2 respectively. Equation 2.8 provides the undistorted image coordinates  $[u', v']^T$  from the distorted image

coordinates  $[u_d, v_d]^T$  under the lens distortion model. Equation 2.8 is expanded and separated into the individual coordinate components in Equations 5.3 and 5.4. Please note that two sets of undistorted image coordinates of the calibration points are obtained from Equations 5.1, 5.2, 5.3 and 5.4. One set is calculated from the world coordinates of the calibration points (Equations 5.1 and 5.2) provided by the pinhole camera model. This set represents theoretical 2D location of the pure perspective projection of the calibration point on the image plane in the absence of lens distortion. The second set is provided by the backward lens distortion model and it is calculated from the distorted image coordinates (Equations 5.3 and 5.4). In a typical calibration scenario, the distorted image coordinates are experimentally determined by processing an acquired image of a calibration point. In the absence of measurement errors and modeling discrepancies, the Cartesian pairs in both of the sets will be coincident in undistorted space. In practice the image point locations provided by the two sets will not be coincident. The proposed calibration technique computes the camera parameters that minimize the discrepancy between the two sets of image points through the nonlinear Newton-Gauss numerical technique. In quantitative terms, the discrepancy is represented by the geometric distances between these two sets of undistorted image coordinates (Equations 5.5, 5.6). Each point correspondence between the object space and the distorted image space constitutes a pair of equations as presented in Equations 5.5 and 5.6. If  $n$  point correspondences are available ( $n \geq 7$ ),  $2n$  number of equations can be formulated; i.e.,  $[f_{1u}, f_{1v}, f_{2u}, f_{2v}, \dots, f_{nu}, f_{nv}]$ . In addition, one more equation will be required to enforce the unit quaternion constraint on the rotation representation as explained in Section 4.1; i.e.,  $f_c = d^2 + a^2 + b^2 + c^2 - 1$ . Altogether the general calibration problem becomes an overdetermined system of  $(2n+1)$  nonlin-



ear equations given  $n$  point correspondences (Equation 5.7) involving fifteen unknown camera parameters (Table 2.1). Fourteen of these parameters are independent. Therefore, at least seven point correspondences are required to generate fifteen equations to solve the calibration problem. The objective of the Newton-Gauss technique is to estimate these parameters by the nonlinear least squares analysis of the system of nonlinear equations (Equation 5.7). It is necessary to calculate the Jacobian matrix (Equation 5.8) of the system of nonlinear equations for the least squares analysis. The generic elements of the Jacobian matrix are presented in Equations 5.9-5.53.

$$u = \frac{sf((d^2 + a^2 - b^2 - c^2)x_w + (2ab - 2cd)y_w + (2ca + 2bd)z_w + t_x) + u_0((2ca - 2bd)x_w + (2bc + 2ad)y_w + (d^2 - a^2 - b^2 + c^2)z_w + t_x)}{(2ca - 2bd)x_w + (2bc + 2ad)y_w + (d^2 - a^2 - b^2 + c^2)z_w + t_x} \quad (5.1)$$

$$v = \frac{f((2ab + 2cd)x_w + (d^2 - a^2 + b^2 - c^2)y_w + (2bc - 2ad)z_w + t_y) + v_0((2ca - 2bd)x_w + (2bc + 2ad)y_w + (d^2 - a^2 - b^2 + c^2)z_w + t_x)}{(2ca - 2bd)x_w + (2bc + 2ad)y_w + (d^2 - a^2 - b^2 + c^2)z_w + t_x} \quad (5.2)$$

$$u' = u_d + (u_d - u_0)(k_1(u_d^2 - 2u_d u_0 + u_0^2 + v_d^2 - 2v_d v_0 + v_0^2) + k_2(u_d^2 - 2u_d u_0 + u_0^2 + v_d^2 - 2v_d v_0 + v_0^2)^2) + 2p_1(u_d - u_0)(v_d - v_0) + p_2(u_d^2 - 2u_d u_0 + u_0^2 + v_d^2 - 2v_d v_0 + v_0^2 + 2(u_d - u_0)^2) \quad (5.3)$$

$$v' = v_d + (v_d - v_0)(k_1(u_d^2 - 2u_d u_0 + u_0^2 + v_d^2 - 2v_d v_0 + v_0^2) + k_2(u_d^2 - 2u_d u_0 + u_0^2 + v_d^2 - 2v_d v_0 + v_0^2)^2) + p_1(u_d^2 - 2u_d u_0 + u_0^2 + v_d^2 - 2v_d v_0 + v_0^2 + 2(v_d - v_0)^2) + 2p_2(u_d - u_0)(v_d - v_0) \quad (5.4)$$

$$f_u = u - u' \quad (5.5)$$

$$f_v = v - v' \quad (5.6)$$

$$F_c = \begin{bmatrix} f_{1u} \\ f_{1v} \\ f_{2u} \\ f_{2v} \\ \dots \\ f_{nu} \\ f_{nv} \\ f_c \end{bmatrix} \quad (5.7)$$

$$J = \begin{bmatrix} \frac{\partial f_{1u}}{\partial s} & \frac{\partial f_{1u}}{\partial f} & \frac{\partial f_{1u}}{\partial k_1} & \frac{\partial f_{1u}}{\partial k_2} & \frac{\partial f_{1u}}{\partial p_1} & \frac{\partial f_{1u}}{\partial p_2} & \frac{\partial f_{1u}}{\partial u_0} & \frac{\partial f_{1u}}{\partial v_0} & \frac{\partial f_{1u}}{\partial d} & \frac{\partial f_{1u}}{\partial a} & \frac{\partial f_{1u}}{\partial b} & \frac{\partial f_{1u}}{\partial c} & \frac{\partial f_{1u}}{\partial t_x} & \frac{\partial f_{1u}}{\partial t_y} & \frac{\partial f_{1u}}{\partial t_z} \\ \frac{\partial f_{1v}}{\partial s} & \frac{\partial f_{1v}}{\partial f} & \frac{\partial f_{1v}}{\partial k_1} & \frac{\partial f_{1v}}{\partial k_2} & \frac{\partial f_{1v}}{\partial p_1} & \frac{\partial f_{1v}}{\partial p_2} & \frac{\partial f_{1v}}{\partial u_0} & \frac{\partial f_{1v}}{\partial v_0} & \frac{\partial f_{1v}}{\partial d} & \frac{\partial f_{1v}}{\partial a} & \frac{\partial f_{1v}}{\partial b} & \frac{\partial f_{1v}}{\partial c} & \frac{\partial f_{1v}}{\partial t_x} & \frac{\partial f_{1v}}{\partial t_y} & \frac{\partial f_{1v}}{\partial t_z} \\ \dots & \dots & \dots & \dots & \dots & \dots & \dots & \dots & \dots & \dots & \dots & \dots & \dots & \dots & \dots \\ \dots & \dots & \dots & \dots & \dots & \dots & \dots & \dots & \dots & \dots & \dots & \dots & \dots & \dots & \dots \\ \frac{\partial f_{nv}}{\partial s} & \frac{\partial f_{nv}}{\partial f} & \frac{\partial f_{nv}}{\partial k_1} & \frac{\partial f_{nv}}{\partial k_2} & \frac{\partial f_{nv}}{\partial p_1} & \frac{\partial f_{nv}}{\partial p_2} & \frac{\partial f_{nv}}{\partial u_0} & \frac{\partial f_{nv}}{\partial v_0} & \frac{\partial f_{nv}}{\partial d} & \frac{\partial f_{nv}}{\partial a} & \frac{\partial f_{nv}}{\partial b} & \frac{\partial f_{nv}}{\partial c} & \frac{\partial f_{nv}}{\partial t_x} & \frac{\partial f_{nv}}{\partial t_y} & \frac{\partial f_{nv}}{\partial t_z} \\ \frac{\partial f_{nv}}{\partial s} & \frac{\partial f_{nv}}{\partial f} & \frac{\partial f_{nv}}{\partial k_1} & \frac{\partial f_{nv}}{\partial k_2} & \frac{\partial f_{nv}}{\partial p_1} & \frac{\partial f_{nv}}{\partial p_2} & \frac{\partial f_{nv}}{\partial u_0} & \frac{\partial f_{nv}}{\partial v_0} & \frac{\partial f_{nv}}{\partial d} & \frac{\partial f_{nv}}{\partial a} & \frac{\partial f_{nv}}{\partial b} & \frac{\partial f_{nv}}{\partial c} & \frac{\partial f_{nv}}{\partial t_x} & \frac{\partial f_{nv}}{\partial t_y} & \frac{\partial f_{nv}}{\partial t_z} \\ \frac{\partial f_c}{\partial s} & \frac{\partial f_c}{\partial f} & \frac{\partial f_c}{\partial k_1} & \frac{\partial f_c}{\partial k_2} & \frac{\partial f_c}{\partial p_1} & \frac{\partial f_c}{\partial p_2} & \frac{\partial f_c}{\partial u_0} & \frac{\partial f_c}{\partial v_0} & \frac{\partial f_c}{\partial d} & \frac{\partial f_c}{\partial a} & \frac{\partial f_c}{\partial b} & \frac{\partial f_c}{\partial c} & \frac{\partial f_c}{\partial t_x} & \frac{\partial f_c}{\partial t_y} & \frac{\partial f_c}{\partial t_z} \end{bmatrix} \quad (5.8)$$

$$\frac{\partial f_u}{\partial s} = \frac{f((d^2 + a^2 - b^2 - c^2)x_w + (2ab - 2cd)y_w + (2ca + 2bd)z_w + t_x)}{(2ca - 2bd)x_w + (2bc + 2ad)y_w + (d^2 - a^2 - b^2 + c^2)z_w + t_z} \quad (5.9)$$

$$\frac{\partial f_u}{\partial f} = \frac{s((d^2 + a^2 - b^2 - c^2)x_w + (2ab - 2cd)y_w + (2ca + 2bd)z_w + t_x)}{(2ca - 2bd)x_w + (2bc + 2ad)y_w + (d^2 - a^2 - b^2 + c^2)z_w + t_z} \quad (5.10)$$

$$\frac{\partial f_u}{\partial k_1} = -(u_d - u_0)(u_d^2 - 2u_d u_0 + u_0^2 + v_d^2 - 2v_d v_0 + v_0^2) \quad (5.11)$$

$$\frac{\partial f_u}{\partial k_2} = -(u_d - u_0)(u_d^2 - 2u_d u_0 + u_0^2 + v_d^2 - 2v_d v_0 + v_0^2)^2 \quad (5.12)$$

$$\frac{\partial f_u}{\partial p_1} = -(2u_d - 2u_0)(v_d - v_0) \quad (5.13)$$

$$\frac{\partial f_u}{\partial p_2} = -u_d^2 + 2u_d u_0 - u_0^2 - v_d^2 + 2v_d v_0 - v_0^2 - 2(u_d - u_0)^2 \quad (5.14)$$

$$\begin{aligned} \frac{\partial f_u}{\partial u_0} = & 1 + k_1(u_d^2 - 2u_d u_0 + u_0^2 + v_d^2 - 2v_d v_0 + v_0^2) + k_2(u_d^2 - 2u_d u_0 + u_0^2 + v_d^2 - 2v_d v_0 + v_0^2)^2 - (u_d - u_0)(k_1(-2u_d + 2u_0) \\ & + 2k_2(u_d^2 - 2u_d u_0 + u_0^2 + v_d^2 - 2v_d v_0 + v_0^2)(-2u_d + 2u_0)) + 2p_1(v_d - v_0) - p_2(-6u_d + 6u_0) \end{aligned} \quad (5.15)$$

$$\frac{\partial f_u}{\partial v_0} = -(u_d - u_0)(k_1(-2v_d + 2v_0) + 2k_2(u_d^2 - 2u_d u_0 + u_0^2 + v_d^2 - 2v_d v_0 + v_0^2)(-2v_d + 2v_0)) + 2p_1(u_d - u_0) - p_2(-2v_d + 2v_0) \quad (5.16)$$

$$\frac{\partial f_u}{\partial d} = \frac{sf(2dx_w - 2cy_w + 2bz_w) + u_0(-2bx_w + 2ay_w + 2dz_w)}{(2ca - 2bd)x_w + (2bc + 2ad)y_w + (d^2 - a^2 - b^2 + c^2)z_w + t_z} - \frac{sf((d^2 + a^2 - b^2 - c^2)x_w + (2ab - 2cd)y_w + (2ca + 2bd)z_w + t_x) + u_0((2ca - 2bd)x_w + (2bc + 2ad)y_w + (d^2 - a^2 - b^2 + c^2)z_w + t_z)(-2bx_w + 2ay_w + 2dz_w)}{((2ca - 2bd)x_w + (2bc + 2ad)y_w + (d^2 - a^2 - b^2 + c^2)z_w + t_z)^2} \quad (5.17)$$

$$\frac{\partial f_u}{\partial a} = \frac{sf(2ax_w + 2by_w + 2cz_w) + u_0(2cx_w + 2dy_w - 2az_w)}{(2ca - 2bd)x_w + (2bc + 2ad)y_w + (d^2 - a^2 - b^2 + c^2)z_w + t_z} - \frac{sf((d^2 + a^2 - b^2 - c^2)x_w + (2ab - 2cd)y_w + (2ca + 2bd)z_w + t_x) + u_0((2ca - 2bd)x_w + (2bc + 2ad)y_w + (d^2 - a^2 - b^2 + c^2)z_w + t_z)(2cx_w + 2dy_w - 2az_w)}{((2ca - 2bd)x_w + (2bc + 2ad)y_w + (d^2 - a^2 - b^2 + c^2)z_w + t_z)^2} \quad (5.18)$$

$$\frac{\partial f_u}{\partial b} = \frac{sf(-2bx_w + 2ay_w + 2dz_w) + u_0(-2dx_w + 2cy_w - 2bz_w)}{(2ca - 2bd)x_w + (2bc + 2ad)y_w + (d^2 - a^2 - b^2 + c^2)z_w + t_z} - \frac{sf((d^2 + a^2 - b^2 - c^2)x_w + (2ab - 2cd)y_w + (2ca + 2bd)z_w + t_x) + u_0((2ca - 2bd)x_w + (2bc + 2ad)y_w + (d^2 - a^2 - b^2 + c^2)z_w + t_z)(-2dx_w + 2cy_w - 2bz_w)}{((2ca - 2bd)x_w + (2bc + 2ad)y_w + (d^2 - a^2 - b^2 + c^2)z_w + t_z)^2} \quad (5.19)$$

$$\frac{\partial f_u}{\partial c} = \frac{sf(-2cx_w - 2dy_w + 2az_w) + u_0(2ax_w + 2by_w + 2cz_w)}{(2ca - 2bd)x_w + (2bc + 2ad)y_w + (d^2 - a^2 - b^2 + c^2)z_w + t_z} - \frac{(sf(d^2 + a^2 - b^2 - c^2)x_w + (2ab - 2cd)y_w + (2ca + 2bd)z_w + t_x) + u_0((2ca - 2bd)x_w + (2bc + 2ad)y_w + (d^2 - a^2 - b^2 + c^2)z_w + t_z)(2ax_w + 2by_w + 2cz_w)}{((2ca - 2bd)x_w + (2bc + 2ad)y_w + (d^2 - a^2 - b^2 + c^2)z_w + t_z)^2} \quad (5.20)$$

$$\frac{\partial f_u}{\partial t_x} = \frac{sf}{(2ca - 2bd)x_w + (2bc + 2ad)y_w + (d^2 - a^2 - b^2 + c^2)z_w + t_z} \quad (5.21)$$

$$\frac{\partial f_u}{\partial t_y} = 0 \quad (5.22)$$

$$\frac{\partial f_u}{\partial t_z} = \frac{u_0}{(2ca - 2bd)x_w + (2bc + 2ad)y_w + (d^2 - a^2 - b^2 + c^2)z_w + t_z} - \frac{sf((d^2 + a^2 - b^2 - c^2)x_w + (2ab - 2cd)y_w + (2ca + 2bd)z_w + t_x) + u_0((2ca - 2bd)x_w + (2bc + 2ad)y_w + (d^2 - a^2 - b^2 + c^2)z_w + t_z)}{((2ca - 2bd)x_w + (2bc + 2ad)y_w + (d^2 - a^2 - b^2 + c^2)z_w + t_z)^2} \quad (5.23)$$



$$\frac{\partial f_v}{\partial s} = 0 \quad (5.24)$$

$$\frac{\partial f_v}{\partial f} = \frac{(2ab + 2cd)x_w + (d^2 - a^2 + b^2 - c^2)y_w + (2bc - 2ad)z_w + t_y}{(2ca - 2bd)x_w + (2bc + 2ad)y_w + (d^2 - a^2 - b^2 + c^2)z_w + t_z} \quad (5.25)$$

$$\frac{\partial f_v}{\partial k_1} = -(v_d - v_0)(u_d^2 - 2u_d u_0 + u_0^2 + v_d^2 - 2v_d v_0 + v_0^2) \quad (5.26)$$

$$\frac{\partial f_v}{\partial k_2} = -(v_d - v_0)(u_d^2 - 2u_d u_0 + u_0^2 + v_d^2 - 2v_d v_0 + v_0^2)^2 \quad (5.27)$$

$$\frac{\partial f_v}{\partial p_1} = -u_d^2 + 2u_d u_0 - u_0^2 - v_d^2 + 2v_d v_0 - v_0^2 - 2(v_d - v_0)^2 \quad (5.28)$$

$$\frac{\partial f_v}{\partial p_2} = -(2u_d - 2u_0)(v_d - v_0) \quad (5.29)$$

$$\frac{\partial f_v}{\partial u_0} = -(v_d - v_0)(k_1(-2u_d + 2u_0) + 2k_2(u_d^2 - 2u_d u_0 + u_0^2 + v_d^2 - 2v_d v_0 + v_0^2)(-2u_d + 2u_0)) - p_1(-2u_d + 2u_0) + 2p_2(v_d - v_0) \quad (5.30)$$

$$\begin{aligned} \frac{\partial f_v}{\partial v_0} = & 1 + k_1(u_d^2 - 2u_d u_0 + u_0^2 + v_d^2 - 2v_d v_0 + v_0^2) + k_2(u_d^2 - 2u_d u_0 + u_0^2 + v_d^2 - 2v_d v_0 + v_0^2)^2 - (v_d - v_0)(k_1(-2v_d + 2v_0) \\ & + 2k_2(u_d^2 - 2u_d u_0 + u_0^2 + v_d^2 - 2v_d v_0 + v_0^2)(-2v_d + 2v_0)) - p_1(-6v_d + 6v_0) + 2p_2(u_d - u_0) \end{aligned} \quad (5.31)$$

$$\begin{aligned} \frac{\partial f_v}{\partial d} = & \frac{f(2cx_w + 2dy_w - 2az_w) + v_0(-2bx_w + 2ay_w + 2dz_w)}{(2ca - 2bd)x_w + (2bc + 2ad)y_w + (d^2 - a^2 - b^2 + c^2)z_w + t_z} \\ & - \frac{f((2ab + 2cd)x_w + (d^2 - a^2 + b^2 - c^2)y_w + (2bc - 2ad)z_w + t_y) + v_0((2ca - 2bd)x_w + (2bc + 2ad)y_w + (d^2 - a^2 - b^2 + c^2)z_w + t_z)(-2bx_w + 2ay_w + 2dz_w)}{((2ca - 2bd)x_w + (2bc + 2ad)y_w + (d^2 - a^2 - b^2 + c^2)z_w + t_z)^2} \end{aligned} \quad (5.32)$$

$$\begin{aligned} \frac{\partial f_v}{\partial a} = & \frac{f(2bx_w - 2ay_w - 2dz_w) + v_0(2cx_w + 2dy_w - 2az_w)}{(2ca - 2bd)x_w + (2bc + 2ad)y_w + (d^2 - a^2 - b^2 + c^2)z_w + t_z} \\ & - \frac{f((2ab + 2cd)x_w + (d^2 - a^2 + b^2 - c^2)y_w + (2bc - 2ad)z_w + t_y) + v_0((2ca - 2bd)x_w + (2bc + 2ad)y_w + (d^2 - a^2 - b^2 + c^2)z_w + t_z)(2cx_w + 2dy_w - 2az_w)}{((2ca - 2bd)x_w + (2bc + 2ad)y_w + (d^2 - a^2 - b^2 + c^2)z_w + t_z)^2} \end{aligned} \quad (5.33)$$

$$\begin{aligned} \frac{\partial f_v}{\partial b} &= \frac{f(2ax_w + 2by_w + 2cz_w) + v_0(-2dx_w + 2cy_w - 2bz_w)}{(2ca - 2bd)x_w + (2bc + 2ad)y_w + (d^2 - a^2 - b^2 + c^2)z_w + t_z} \\ &\quad - \frac{f((2ab + 2cd)x_w + (d^2 - a^2 + b^2 - c^2)y_w + (2bc - 2ad)z_w + t_y) + v_0((2ca - 2bd)x_w + (2bc + 2ad)y_w + (d^2 - a^2 - b^2 + c^2)z_w + t_z)(-2dx_w + 2cy_w - 2bz_w)}{((2ca - 2bd)x_w + (2bc + 2ad)y_w + (d^2 - a^2 - b^2 + c^2)z_w + t_z)^2} \end{aligned} \quad (5.34)$$

$$\begin{aligned} \frac{\partial f_v}{\partial c} &= \frac{f(2dx_w - 2cy_w + 2bz_w) + v_0(2ax_w + 2by_w + 2cz_w)}{(2ca - 2bd)x_w + (2bc + 2ad)y_w + (d^2 - a^2 - b^2 + c^2)z_w + t_z} \\ &\quad - \frac{f((2ab + 2cd)x_w + (d^2 - a^2 + b^2 - c^2)y_w + (2bc - 2ad)z_w + t_y) + v_0((2ca - 2bd)x_w + (2bc + 2ad)y_w + (d^2 - a^2 - b^2 + c^2)z_w + t_z)(2ax_w + 2by_w + 2cz_w)}{(2ca - 2bd)x_w + (2bc + 2ad)y_w + (d^2 - a^2 - b^2 + c^2)z_w + t_z)^2} \end{aligned} \quad (5.35)$$

$$\frac{\partial f_v}{\partial t_x} = 0 \quad (5.36)$$

$$\frac{\partial f_v}{\partial t_y} = \frac{f}{(2ca - 2bd)x_w + (2bc + 2ad)y_w + (d^2 - a^2 - b^2 + c^2)z_w + t_z} \quad (5.37)$$

$$\begin{aligned} \frac{\partial f_v}{\partial t_z} &= \frac{v_0}{(2ca - 2bd)x_w + (2bc + 2ad)y_w + (d^2 - a^2 - b^2 + c^2)z_w + t_z} \\ &\quad - \frac{f((2ab + 2cd)x_w + (d^2 - a^2 + b^2 - c^2)y_w + (2bc - 2ad)z_w + t_y) + v_0((2ca - 2bd)x_w + (2bc + 2ad)y_w + (d^2 - a^2 - b^2 + c^2)z_w + t_z)}{((2ca - 2bd)x_w + (2bc + 2ad)y_w + (d^2 - a^2 - b^2 + c^2)z_w + t_z)^2} \end{aligned} \quad (5.38)$$

$$\frac{\partial f_c}{\partial s} = 0 \quad (5.39)$$

$$\frac{\partial f_c}{\partial f} = 0 \quad (5.40)$$

$$\frac{\partial f_c}{\partial k_1} = 0 \quad (5.41)$$

$$\frac{\partial f_c}{\partial k_2} = 0 \quad (5.42)$$

$$\frac{\partial f_c}{\partial p_1} = 0 \quad (5.43)$$

$$\frac{\partial f_c}{\partial p_2} = 0 \quad (5.44)$$

$$\frac{\partial f_c}{\partial u_0} = 0 \quad (5.45)$$

$$\frac{\partial f_c}{\partial v_0} = 0 \quad (5.46)$$

$$\frac{\partial f_c}{\partial d} = 2d \quad (5.47)$$

$$\frac{\partial f_c}{\partial a} = 2a \quad (5.48)$$

$$\frac{\partial f_c}{\partial b} = 2b \quad (5.49)$$

$$\frac{\partial f_c}{\partial c} = 2c \quad (5.50)$$

$$\frac{\partial f_c}{\partial t_x} = 0 \quad (5.51)$$

$$\frac{\partial f_c}{\partial t_y} = 0 \quad (5.52)$$

$$\frac{\partial f_c}{\partial t_z} = 0 \quad (5.53)$$

The termination criterion for the nonlinear iterative search has yet to be determined. The Newton-Gauss nonlinear least squares analysis proceeds iteratively starting with the initial estimation of the camera parameters  $C_{initial}$  (Section 5.2) to find the optimal camera parameters  $C_{optimal}$  that minimize the modeling discrepancy. In each iteration of the Newton-Gauss algorithm, the current camera parameters  $C_{current}$  are corrected by a correction vector  $\beta$  which can be calculated as follows:

$$\beta = (J^T \times J)^{-1} \times (-J^T \times F_c) \quad (5.54)$$

The camera parameters are updated after each iteration (Equation 5.55).

$$C_{current} = \beta + C_{current} \quad (5.55)$$

In practice computing the inverse of the matrix product  $(J^T \times J)^{-1}$  is computationally inefficient. The system of linear equations provided by Equation 5.54 has the coefficient matrix  $(J^T \times J)$  and the constant vector  $(-J^T \times F_c)$ . Equation 5.54 can be solved without computing the inverse of the coefficient matrix. Computationally a more efficient alternative is the Gaussian elimination of the coefficient matrix and the constant vector. Since  $(J^T \times J)$  is a positive definite matrix, it is preferable to decompose it into the product of a lower triangular matrix and its conjugate transpose by applying Cholesky's method. Because, it is relatively easy to solve the Gaussian elimination problem when the coefficient matrix is either upper or lower triangular matrix. Let  $(J^T \times J)$  be decomposed into a lower triangular matrix  $L$  and its conjugate transpose  $L^*$  by Cholesky's method; i.e.,  $(J^T \times J) = LL^*$ . Substituting the decomposed coefficient matrix into Equation 5.54 yields:

$$LL^*\beta = -J^T \times F_c \quad (5.56)$$

Equation 5.56 can be solved in two steps. At first, Gaussian elimination of the matrix  $L$  and the vector  $(-J^T \times F_c)$  provides  $L^*\beta = y$ . Subsequently, the Gaussian elimination of  $L^*$  and  $y$  produces the correction vector  $\beta$ .

The Euclidean norm of the correction vector  $\|\beta\|$  can be used as one of the termination criteria.  $\|\beta\|$  is defined in Equation 5.57. When  $\|\beta\|$  is very close to zero; i.e.,  $\|\beta\| \cong 0$ , the camera parameters can be considered as optimal. Some predefined threshold  $\epsilon_{min}$  is chosen to quantify  $\|\beta\|$  being in the vicinity of zero. Usually the



value of  $\epsilon_{min}$  is chosen such that  $\epsilon_{min} \approx 0$ . When  $\|\beta\| > \epsilon_{min}$ , the iterative search continues to obtain a better estimate of the camera parameters. Otherwise, the search terminates and the current camera parameters are considered to be optimal; i.e.,  $C_{optimal} = C_{current}$  and it is said that the iterative search converged to a solution. Convergence to a solution largely depends on the quality of the calibration data (i.e., point correspondences). If the calibration is attempted with noisy data, a solution is not likely to be found. The iterative search will continue indefinitely and  $\|\beta\|$  will never come close to zero. In order to avoid this situation, a maximum number of iterations  $i_{max}$  is defined. When the number of iterations  $i$  in the search exceeds  $i_{max}$  ( $i > i_{max}$ ), the loop terminates and it is said that a solution is not found.

$$\|\beta\| = \sqrt{\beta_1^2 + \beta_2^2 + \beta_3^2 + \cdots + \beta_{14}^2 + \beta_{15}^2} \quad (5.57)$$

## 5.2 Initial Approximation

Convergence of the calibration algorithm to an optimal solution (i.e., global minimum) largely depends on a good initial approximation of the camera parameters. Prior to calibration, very little information is usually available about the true values of the parameters. Hence an initial approximation must be acquired from the readily available calibration data (i.e., point correspondences). In addition to the calibration data, the manufacturer of the camera also provides nominal values for some of the camera parameters pertaining to the geometry and optics of the camera; i.e., nominal focal length and pixel spacing on the sensor element. The nominal focal length of the camera provided by the manufacturer is usually a good approximation of the effective focal length  $f$  for the nonlinear iterative search. The ratio of the pixel spacing length

in the horizontal and vertical directions provides a good initial guess for the scale factor  $s$ . Usually the lens distortion coefficients (radial and tangential) are very small and they can be initially approximated as zero. All the intrinsic camera parameters except the image center  $(u_0, v_0)$  can thus be approximated from information provided by the camera manufacturer. The geometrical center of the rectangular image sensor is a good initial approximation of the image center coordinates. Approximating the intrinsic parameters from the information provided by the manufacturer reduces computational cost significantly. The extrinsic camera parameters have yet to be approximated with acceptable accuracy. Available calibration data can be used to approximate the extrinsic camera parameters through linear least squares analysis.

Linear least squares analysis was used to estimate the camera parameters in [17]. The underlying camera model in [17] was linear since it did not account for lens distortion. As a result, the estimates of the camera parameters from [17] are usually not as accurate as expected. This technique is however attractive in applications where the emphasis is on computation cost and robustness rather than accuracy. A solution is always guaranteed by this technique since there is no issue with convergence. In later years this method was used to obtain an initial approximation of the camera parameters for many iterative nonlinear camera calibration techniques; e.g., [4], [19]. In the proposed calibration technique, approximations of the extrinsic orientative and position parameters can be estimated by linear least squares analysis. A simple algorithm in [29] was presented for estimating the extrinsic parameters. This algorithm was adopted here with minor modification to obtain the initial approximations of the extrinsic parameters for the nonlinear iterative search.

### 5.2.1 Estimating Extrinsic Parameters

Given the distorted image coordinates  $[u_d, v_d]^T$  and the corresponding world coordinates  $[x_w, y_w, z_w]^T$  of a calibration point, the general problem is to find the orientation and position of the world coordinate frame with respect to the camera coordinate frame. Since lens distortion effects are neglected, the image points can be considered to be produced by ray tracing in a pinhole camera model; i.e., image coordinates are considered to be generated by pure perspective projection. With only six unknown extrinsic parameters, the availability of enough point correspondences presents an opportunity to approach the problem as an overdetermined system of linear equations. A solution can thus be found by linear least squares analysis.

### 5.2.2 Linear Least Squares Analysis

Readily available image coordinates  $[u_d, v_d]^T$  can be translated to a coordinate frame with origin at the geometrical center of the image sensor (i.e., the initial approximation of the image center). Essentially this coordinate frame has the same orientation as the image coordinate frame. Let the translated image coordinates be  $[x_i, y_i]^T$ . The geometrical center of the image sensor is located at  $[u'_0, v'_0]^T$  with respect to the image coordinate frame.

$$\begin{bmatrix} x_i \\ y_i \end{bmatrix} = \begin{bmatrix} u_d - u'_0 \\ v_d - v'_0 \end{bmatrix} \quad (5.58)$$

If the scale factor is assumed to be unity, dividing Equation 2.2 by 2.4 provides the following relationship:

$$\frac{x_i}{y_i} = \frac{x_c}{y_c} \quad (5.59)$$



Note that, Equation 5.59 is independent of the focal length which is particularly advantageous since some of the extrinsic parameters can be estimated irrespective of focal length from this relationship. The general transformation of the position vectors from the world coordinate frame to the camera coordinate frame can be formulated as follows:

$$\begin{bmatrix} x_c \\ y_c \\ z_c \end{bmatrix} = {}^cR_w \times \begin{bmatrix} x_w \\ y_w \\ z_w \end{bmatrix} + \begin{bmatrix} t_x \\ t_y \\ t_z \end{bmatrix} \quad (5.60)$$

$$= \begin{bmatrix} r_{11} & r_{12} & r_{13} \\ r_{21} & r_{22} & r_{23} \\ r_{31} & r_{32} & r_{33} \end{bmatrix} \times \begin{bmatrix} x_w \\ y_w \\ z_w \end{bmatrix} + \begin{bmatrix} t_x \\ t_y \\ t_z \end{bmatrix}$$

In Equation 5.60 the rotation matrix  ${}^cR_w$  defines the orientation of the world coordinate frame with respect to the camera coordinate frame and  $[t_x, t_y, t_z]^T$  is the position vector of the origin of the world coordinate frame with respect to the camera coordinate frame. Expanding Equation 5.60 produces the following:

$$x_c = r_{11}x_w + r_{12}y_w + r_{13}z_w + t_x \quad (5.61)$$

$$y_c = r_{21}x_w + r_{22}y_w + r_{23}z_w + t_y \quad (5.62)$$

$$z_c = r_{31}x_w + r_{32}y_w + r_{33}z_w + t_z \quad (5.63)$$



Substituting Equations 5.61 and 5.62 into Equation 5.59 and subsequent algebraic manipulations yield the following linear equation:

$$\begin{aligned}\frac{x_i}{y_i} &= \frac{r_{11}x_w + r_{12}y_w + r_{13}z_w + t_x}{r_{21}x_w + r_{22}y_w + r_{23}z_w + t_y} \\ \Rightarrow r_{11}x_w y_i + r_{12}y_w y_i + r_{13}z_w y_i + t_x y_i &= r_{21}x_w x_i + r_{22}y_w x_i + r_{23}z_w x_i + t_y x_i \\ \Rightarrow r_{11}x_w y_i + r_{12}y_w y_i + r_{13}z_w y_i + t_x y_i - r_{21}x_w x_i - r_{22}y_w x_i - r_{23}z_w x_i - t_y x_i &= 0\end{aligned}\tag{5.64}$$

Equation 5.64 can be written in matrix form:

$$\begin{bmatrix} x_w y_i & y_w y_i & z_w y_i & y_i & -x_w x_i & -y_w x_i & -z_w x_i & -x_i \end{bmatrix} \times \begin{bmatrix} r_{11} & r_{12} & r_{13} & t_x & r_{21} & r_{22} & r_{23} & t_y \end{bmatrix}^T = 0 \tag{5.65}$$

Equation 5.65 is a homogeneous linear equation in eight unknowns:

$$r_{11}, r_{12}, r_{13}, r_{21}, r_{22}, r_{23}, t_x, t_y.$$

For each point correspondence between the object space and the image space, one such equation (Equation 5.65) can be derived and a system of linear equations can be formed. The elements of the coefficient matrix of this system of linear equations are the products of the components of image and world coordinates. Hence, calculating the coefficient matrix is a trivial computational problem. If  $n$  point correspondences are available and  $n > 7$ , the system of linear equations is overdetermined and can be solved by linear least squares analysis. It should be noted that if a solution to Equation 5.67 exists, any multiple of that solution will also be a solution to Equation 5.67 due to the homogeneous nature of the system. Because of this, it is favorable to convert Equation 5.67 to an inhomogeneous equation (Equation 5.66). Arbitrarily

setting one unknown to any nonzero real value converts the homogeneous system into an inhomogeneous system in the remaining seven unknowns. A unique solution to the inhomogeneous system of linear equations (Equation 5.66) can be estimated by linear least squares analysis. The unknown  $t_y$  is arbitrarily set to unity to realize the conversion of the homogeneous system. The corresponding coefficient  $-x_i$  becomes a constant term in the converted inhomogeneous linear system of equations (Equation 5.66):

$$\begin{bmatrix} x_w y_i & (y_w + \delta_y) y_i & z_w y_i & y_i & -x_w x_i & -(y_w + \delta_y) x_i & -z_w x_i \end{bmatrix} \times \begin{bmatrix} r_{11} & r_{12} & r_{13} & t_x & r_{21} & r_{22} & r_{33} \end{bmatrix}^T = x_i \quad (5.66)$$

If the actual value of  $t_y$  is close to zero, the coefficient matrix of the linear system of equations will be poorly scaled. Offsetting the  $y_w$  component by some arbitrary value  $\delta_y$ , will result in a translated version of the linear system presented in Equation 5.65. This translation will effectively avoid the possible poor scaling of the coefficient matrix. Due to this offset, the  $y$  component of the position vector of the origin of the world coordinate frame with respect to the camera coordinate frame will take a new value; i.e.,  $t_{y'} = t_y + \delta_y$ . Consequently Equation 5.65 will become:

$$\begin{bmatrix} x_w y_i & (y_w + \delta_y) y_i & z_w y_i & y_i & -x_w x_i & -(y_w + \delta_y) x_i & -z_w x_i & -x_i \end{bmatrix} \times \begin{bmatrix} r_{11} & r_{12} & r_{13} & t_x & r_{21} & r_{22} & r_{33} & t_{y'} \end{bmatrix}^T = 0 \quad (5.67)$$

Equation 5.66 is a system of linear equations of the form  $\mathbf{X} \times \mathbf{B} = \mathbf{Y}$ . The solution to this system can be found by applying the pseudo-inverse method:

$$\mathbf{B} = (\mathbf{X}^T \times \mathbf{X})^{-1} \times \mathbf{X}^T \times \mathbf{Y}$$

Let the solution obtained from the least squares analysis of the system of linear equations in Equation 5.66 be:

$$\mathbf{B} = [r'_{11} \ r'_{12} \ r'_{13} \ t'_x \ r'_{21} \ r'_{22} \ r'_{23}]^T$$

$\mathbf{B}^* = [\mathbf{B}, 1] = [r'_{11} \ r'_{12} \ r'_{13} \ t'_x \ r'_{21} \ r'_{22} \ r'_{23} \ 1]^T$  is one of the many solutions to Equation 5.67; i.e., it is merely a scaled version of the sought after solution. One can use the orthonormality property of the rows of a rotation matrix; i.e.,  $r_{21}^2 + r_{22}^2 + r_{23}^2 = 1$ , to determine this scaling factor  $c$ . Subsequently the scaling factor  $c$  of the solution can be estimated from the following equation:

$$c = \frac{1}{\sqrt{r_{21}'^2 + r_{22}'^2 + r_{23}'^2}} \quad (5.68)$$

Once the scaling factor  $c$  is determined estimates of the unknowns  $t_x, t_y$  and the first and second rows of the rotation matrix (i.e.,  $\tilde{t}_x, \tilde{t}_y, \tilde{R}'_1, \tilde{R}'_2$ ) can be calculated using the following equations:

$$\tilde{t}_x = c \cdot t'_x \quad \tilde{t}_y = c \cdot \delta_y \quad (5.69)$$

$$\tilde{R}'_1 = c \cdot [r'_{11} \ r'_{12} \ r'_{13}] \quad \tilde{R}'_2 = c \cdot [r'_{21} \ r'_{22} \ r'_{23}] \quad (5.70)$$

Since there is likely to be an appreciable amount of noise in the available point correspondences, the estimates obtained from the linear least squares analysis will not be highly accurate. Specifically, the estimates of the first and second rows of the rotation matrix will lack orthonormality. These estimates ( i.e.,  $\tilde{R}'_1$  and  $\tilde{R}'_2$ ) need to be further refined in order to calculate a reasonably accurate estimation of the rotation matrix. In order to obtain a good estimate of the first two rows of the rotation matrix, the orthonormality constraint must be enforced on the estimates  $\tilde{R}'_1$  and  $\tilde{R}'_2$ .



### 5.2.3 Forcing Orthonormality

The basic problem is to find two vectors,  $\mathbf{a}'$  and  $\mathbf{b}'$ , that are orthonormal to each other and as close as possible to two given vectors  $\mathbf{a}$  and  $\mathbf{b}$  respectively. The orthonormal vectors can be estimated as follows:

$$\mathbf{a}' = \mathbf{a} + k\mathbf{b} \quad \text{and} \quad \mathbf{b}' = \mathbf{b} + k\mathbf{a} \quad (5.71)$$

$$\mathbf{a}' \cdot \mathbf{b}' = \mathbf{a} \cdot \mathbf{b} + k(\mathbf{a} \cdot \mathbf{a} + \mathbf{b} \cdot \mathbf{b}) + k^2 \mathbf{a} \cdot \mathbf{b} = 0 \quad (5.72)$$

The solution to the quadratic involving  $k$  is usually found to be numerically ill-behaved especially when the coefficient  $\mathbf{a} \cdot \mathbf{b}$  is very small. This occurs because  $\mathbf{a}$  and  $\mathbf{b}$  are already very close to orthonormality. However, assuming  $\mathbf{a} \cdot \mathbf{a}$  and  $\mathbf{b} \cdot \mathbf{b}$  to be near one offers the opportunity to use the following approximate solution:

$$k \approx -\frac{1}{2} \mathbf{a} \cdot \mathbf{b} \quad (5.73)$$

Better estimates of the first two rows of the rotation matrix  $\tilde{R}_1$  and  $\tilde{R}_2$  can be calculated by enforcing orthonormality as explained above:

$$\tilde{R}_1 = \tilde{R}'_1 + k \cdot \tilde{R}'_2 \quad \tilde{R}_2 = \tilde{R}'_2 + k \cdot \tilde{R}'_1 \quad k = -\frac{1}{2} \tilde{R}'_1 \cdot \tilde{R}'_2 \quad (5.74)$$

### 5.2.4 Estimating the Rotation Matrix

In order to recover the full rotation matrix once the estimates of the first two rows have been found, the third row must also be estimated. From the properties of rotation matrices, it is known that the cross product of the first and second row of the rotation matrix will provide the third row. Hence, the estimate of the third row



of the rotation matrix  $\tilde{R}_3$  can be calculated as:

$$\tilde{R}_3 = \tilde{R}_1 \times \tilde{R}_2 \quad (5.75)$$

### 5.2.5 Retrieving the Range of the Target

The range of the target  $t_z$  can be retrieved using either or both of the relationships given in Equations 2.3 and 2.4. Since the scale factor  $s$  is not estimated here, it is more logical in a mathematical sense to use the relationship in Equation 2.4 only. Substituting Equation 5.63 into Equation 2.4 yields the following:

$$\begin{aligned} y_i &= \frac{r_{21}x_w + r_{22}y_w + r_{23}z_w + t_y}{r_{31}x_w + r_{32}y_w + r_{33}z_w + t_z} f \\ \Rightarrow (r_{21}x_w + r_{22}y_w + r_{23}z_w + t_y)f - y_i t_z &= r_{31}x_w y_i + r_{32}y_w y_i + r_{33}z_w y_i \end{aligned} \quad (5.76)$$

Equation 5.76 can be rearranged in matrix form:

$$\begin{bmatrix} r_{21}x_w & r_{22}y_w & r_{23}z_w & t_y & -y_i \end{bmatrix} \times \begin{bmatrix} f & t_z \end{bmatrix}^T = r_{31}x_w y_i + r_{32}y_w y_i + r_{33}z_w y_i \quad (5.77)$$

The estimates  $\tilde{R}_2$ ,  $\tilde{t}_y$  and  $\tilde{R}_3$  can be substituted in Equation 5.77 to solve the over-determined linear system of equations (Equation 5.77) by least squares analysis. A solution to Equation 5.77 will provide the estimates  $\tilde{f}$  and  $\tilde{t}_z$  for the effective focal length  $f$  and the range of the target  $t_z$  respectively.

### 5.2.6 Sign Ambiguity

Sign ambiguity may arise in the estimates from the scale factor  $c$  of the solution to Equation 5.66. It can be easily seen from Equation 5.68 that the scale factor  $c$  will always have a non negative value. However, when the actual value of  $t_y$  is negative, setting  $t_y$  arbitrarily to unity will affect the estimates of  $t_x$ ,  $t_y$ ,  $R_1$  and  $R_2$  sign wise.

The least squares analysis of Equation 5.66 retrieves good estimates of the absolute values of the extrinsic parameters disregarding the sign. Therefore, sign ambiguity must be resolved. Solving Equation 5.77 yields the estimate of the effective focal length  $\tilde{f}$  besides the range of the data  $t_z$ . Though the manufacturer provided nominal focal length is used as the initial approximation of the effective focal length  $f$ , the estimated effective focal length from the least squares solution of Equation 5.77 can be used to resolve the sign ambiguity. If sign ambiguity is present in the approximation, the estimate of the effective focal length from the solution of Equation 5.77 will be negative in sign. When this indication of sign ambiguity is evident, the estimates of  $t_x, t_y, R_1$  and  $R_2$  can be compensated for accordingly by simply multiplying these estimates by  $-1$ .

### 5.3 Calibration Algorithm

The proposed camera calibration technique will be summarized in the form of an algorithm. Refer to Figure 5.1 for a graphical representation of the algorithm. The algorithm is described in the following steps:

- **Step 1:** Input the point correspondences; i.e., world coordinates of the calibration points and their respective image coordinates.

$$(x_w, y_w, z_w) \leftrightarrow (u_d, v_d)$$

- **Step 2:** Choose suitable termination conditions; i.e.,  $i_{max}$  and  $\epsilon_{min}$ .
- **Step 3:** Initialize the camera model as described in Section 5.2.

$$C_{current} = C_{initial}$$

- **Step 4:** Calculate the undistorted image coordinates from the world coordinates by applying pure perspective projection provided by the camera model disregarding lens distortion.

$$(x_w, y_w, z_w) \xrightarrow{\text{Pinhole Camera Model}} (u, v)$$

- **Step 5:** Calculate the undistorted image coordinates from the distorted (measured) image coordinates by applying the lens distortion model. The lens distortion coefficients are provided by the current camera model.

$$(u_d, v_d) \xrightarrow{\text{Lens Distortion Model}} (u', v')$$

- **Step 6:** Evaluate the function vector  $F_c$  (Equation 5.7) and the Jacobian matrix (Equation 5.8) for the current camera model and the point correspondences. Calculate the correction vector  $\beta$  for the camera model applying Equation 5.54.
- **Step 7:** Update the camera model according to Equation 5.55.
- **Step 8:** Check the termination conditions.
  - If  $\|\beta\| < \epsilon_{min}$ , proceed to step 10.
  - If  $i < i_{max}$ , go to step 4 else terminate the loop and go to step 9.
- **Step 9:** No solution is found.
- **Step 10:** A solution is found. The optimal camera model which minimizes error is the current camera model.

$$C_{optimal} = C_{current}$$

The algorithm successfully converges to a solution and terminates.

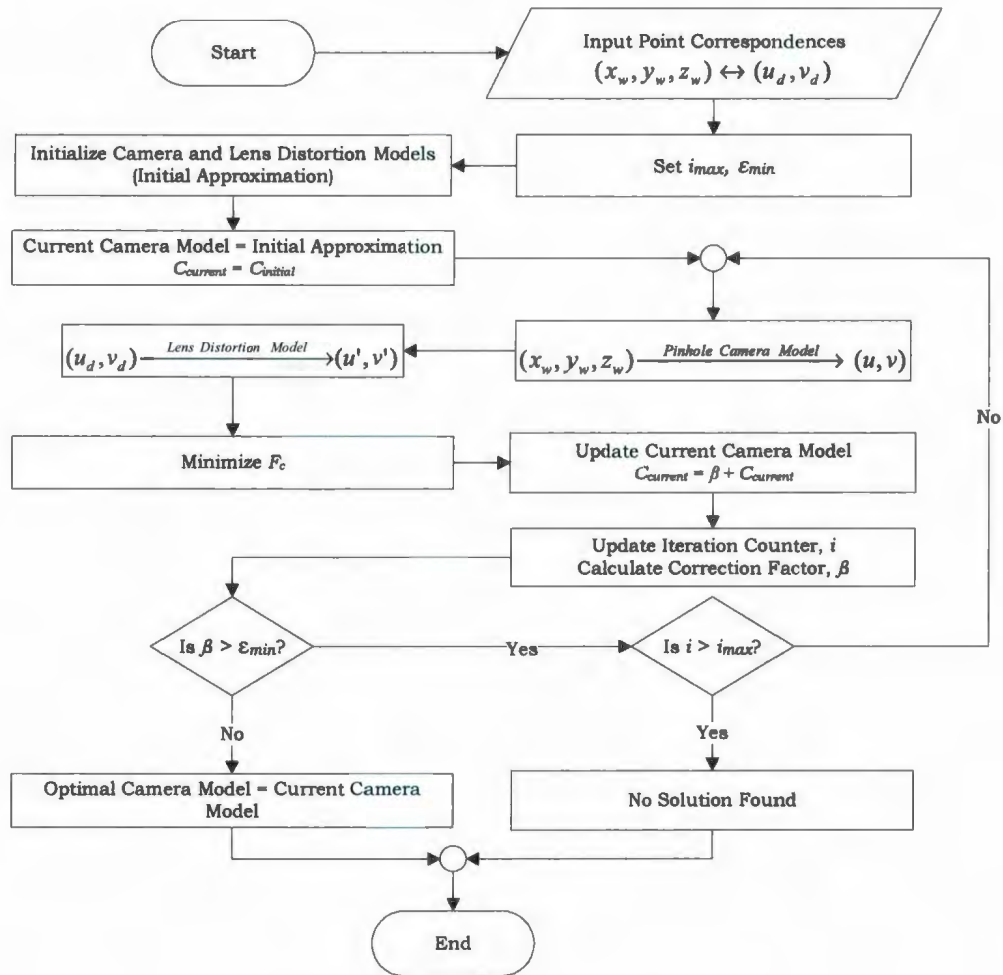


Figure 5.1: Proposed Calibration Algorithm



## 5.4 Implementation of the Algorithm

This algorithm was implemented in MATLAB<sup>®</sup> and the algebraic calculations and manipulations were performed in Maple<sup>™</sup>. The language conversion tool of Maple<sup>™</sup> was used to directly generate the algebraic expressions into MATLAB<sup>®</sup> code. To input the manufacturer provided camera information; i.e., initialize the camera, a MATLAB<sup>®</sup> based *Graphical User Interface (GUI)* was developed (Figure 5.2).

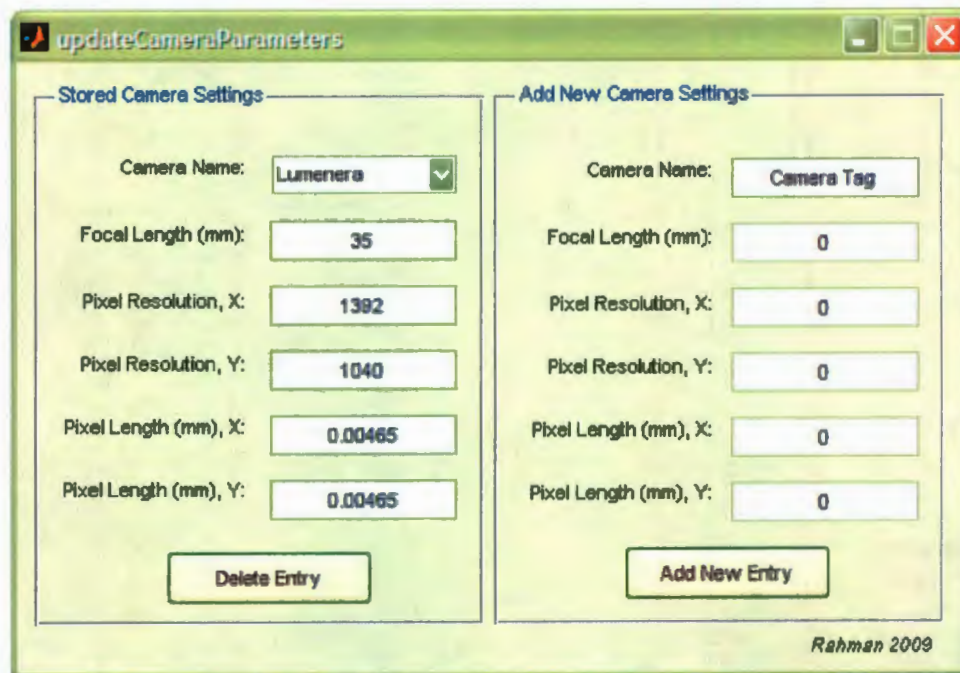


Figure 5.2: Camera Initialization GUI

The calling syntax of the camera calibration routine is:

```
>>calCam('cameraName', inputData);
```

The two arguments of the calibration routine are '*cameraName*' and *inputData*. '*cameraName*' is a unique string that identifies the particular camera. *inputData* is a

matrix of dimension  $n \times 5$ , where the first three columns consist of the world coordinates of the  $n$  calibration points in the order of  $X - Y - Z$ . The last two columns consist of the pixel coordinates of the image of the calibration points in the order of  $U - V$ .

## Chapter 6

# Implementation of the Camera Calibration Technique

Implementing a typical camera calibration technique in machine vision starts with the acquisition of images of a calibration target. The calibration target is an array of calibration points in 3-Space. Depending on the particular camera calibration technique employed it can be coplanar or 3D. Two such widely used calibration targets are shown in Figure 6.1. With respect to some world coordinate frame, positions of these calibration points are known. In addition, the image coordinates of these calibration points are retrieved by analyzing the images of the calibration target by suitable image feature extraction technique. Thus a set of point correspondences is generated between the object space and the image space. The camera calibration technique estimates the camera parameters according to the underlying camera model based on these point correspondences.

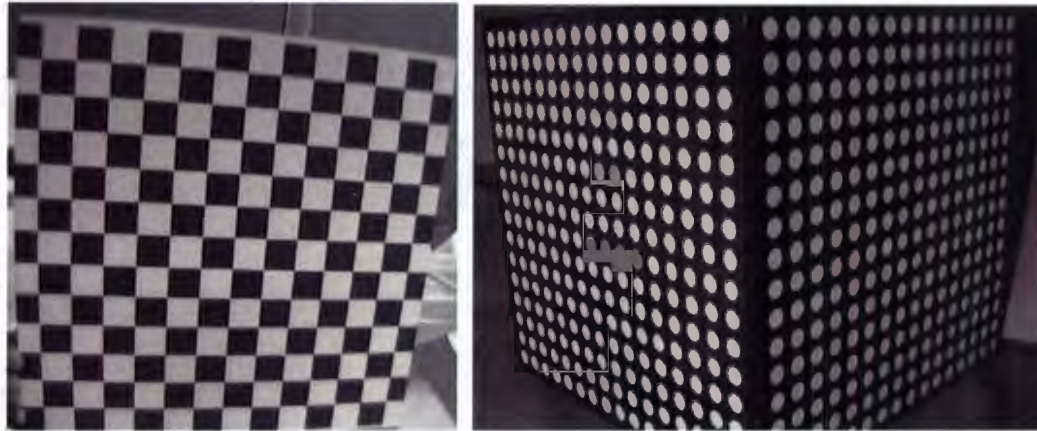


Figure 6.1: Typical Calibration Targets: Checkerboard Pattern (Left) and Dot Pattern (Right)

## 6.1 Conventional Calibration Targets

The most popular calibration targets are arrays of the calibration points arranged either in a checkerboard pattern (e.g. [3, 6, 5]) or in a dot pattern (e.g. [19, 4]). The calibration points are represented by the vertices of the squares in the checkerboard pattern or the centers of the circles in the dot pattern (Figure 6.1). These patterns are usually attached to a planar surface for rigidity to constitute a  $2D$  calibration target. Sometimes multiple planar targets are arranged in a cubic structure comprising a single calibration target which is called the  $3D$  calibration target. The color contrast of the calibration targets is usually black and white to facilitate the image feature extraction technique in localizing the calibration points in the image space.

In the literature reported contemporary calibration targets are co-axial circles [36], spheres ([37, 38]) and arbitrary symmetrical objects [39]. While calibrating a camera using these contemporary targets require relatively less effort and time, this conve-



nience is obtained at the cost of accuracy. One prominent disadvantage of employing the contemporary target is that the image of the target does not cover the entire field of view of the camera. Since the lens distortion coefficients are the functions of the position of the ideal image point on the image plane, these targets fails to estimate the lens distortion coefficients accurately. Contemporary camera calibration techniques such as ([36, 37, 38, 39]) focus on projective geometry rather than Euclidean reconstruction. As a result, when the camera is calibrated for applications especially designed for highly accurate Euclidean reconstruction, these calibration targets fail to meet the accuracy requirement. The contemporary calibration targets used in calibration techniques ([36, 37, 38, 39]) are respectively shown in Figures 6.2(a), 6.2(b), 6.2(c), 6.2(d).

Generating high quality calibration data requires accurate localization of the calibration points in the image space and as well as in the object space. Unfortunately there are a number of disadvantages associated with using the conventional calibration target in this regard. The accuracy of the calibration data in the image space and as well as in the object space can be potentially compromised when these conventional targets are used.

### **6.1.1 Accuracy in the Image Space**

Perspective distortion can be defined as the distortion of the shape in the image of a 3D object or a finite planar surface produced by perspective projection. Perspective projection does not preserve angle, length or ratio of length. Failure to preserve these basic geometrical properties imposes a lot of difficulties on the localization of the

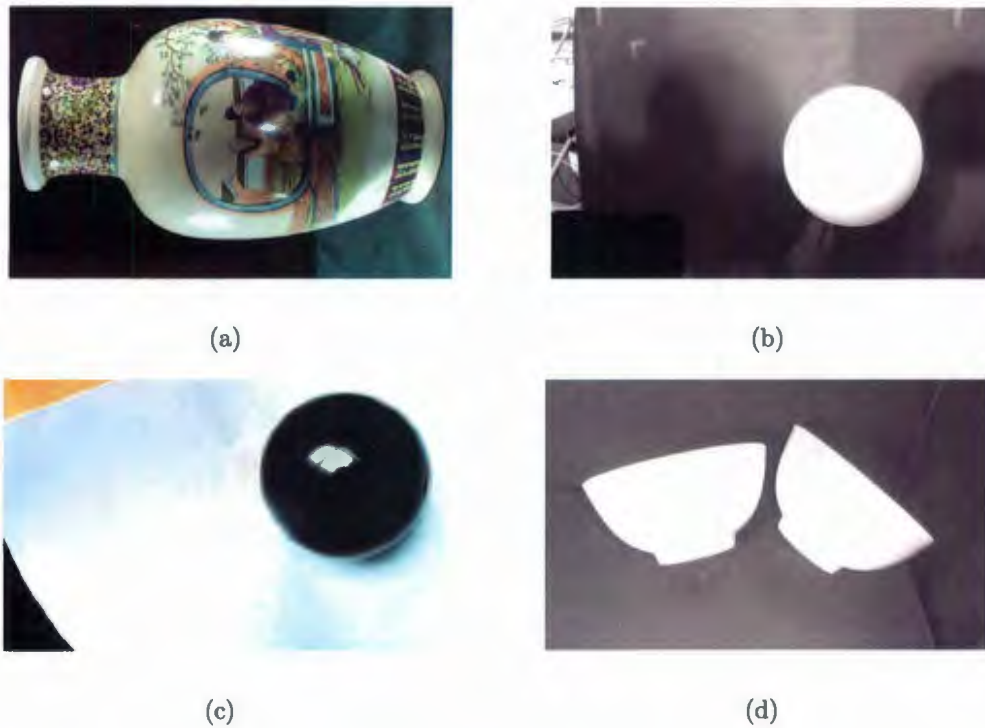


Figure 6.2: Contemporary Calibration Targets

calibration points in the image accurately. These difficulties are unique to each type of the calibration target; i.e., checkerboard pattern and dot pattern.

When the checkerboard pattern is used as the calibration target, the orthogonality of the sides of the squares is not preserved in the image. Moreover, lens distortion causes the straight lines to be imaged as curves (Figure 6.3). As a result, localizing the vertices in the image space becomes a position of fitting a curve along the side of the squares and subsequently finding the intersection points. This problem is virtually unsolvable as the lens distortion parameters are not known at this stage of camera calibration.

The centers of the circles represent the calibration point in the dot patterned

calibration target. The projective image of a circle is an ellipse. The amount of perspective distortion present in the image of a circle primarily depends on two factors. The factors being the relative orientation of the plane on which the circle is lying with respect to the image plane and the relative position of the center of the circle with respect to the optical center of the lens. Perspective distortion becomes practically zero; i.e., the image of a circle is also a circle, when the two planes (i.e., the planes containing the circle and the image) are parallel and the center of the circle (in the object space) coincides with the optical axis of the camera. Perspective distortion increases as the plane encompassing the circle moves further away from being parallel to the image plane. In this case, the increased perspective distortion in the image is manifested as increased eccentricity of the ellipse. In order to localize the calibration point in the image space, it is required to find the projection of the center of the circle. As the distance between the center of the circle and optical axis of the camera increases, the projection of the center of the circle moves further away from the geometrical center of the ellipse in the image space (Figure 6.4). Moreover, the effect of perspective projection on each dot in the pattern is not uniform in the image space due to the different position and orientation of each dot. As a result, it is difficult to localize the calibration points in the image space with acceptable accuracy. The presence of non-uniform perspective distortion in the image of the dot patterned calibration target was corrected mathematically in [4] in order to compensate for this phenomenon.



### 6.1.2 Accuracy in the Object Space

The conventional calibration target is often produced by consumer grade printers which usually can not provide acceptable geometrical accuracy. Assuming that the produced calibration target complies with the geometrical design can yield low quality calibration data. Therefore, it is required to employ metrological means to measure the relative positions of the calibration points in the calibration target. Since these targets are printed on paper, they can not be expected to retain uniform geometrical properties throughout the calibration process due to thermal expansion. The printers discretize the edges of the patterns which results in loss of resolution in the geometric shapes. Despite employing metrological techniques to enhance accuracy in the object space, the inaccuracy in the geometrical shape due to discretization ultimately adds to the noise present in the image.

## 6.2 Generating Highly Accurate Point Correspondences

In this work a novel approach towards generating the point correspondences for camera was adopted. It employs a calibration system (Figure 6.5) that features a spherical calibration target and a calibration rig. The calibration rig is the physical framework for acquisition of the calibration data. The heart of the system is a computer that hosts a MATLAB<sup>®</sup> based *GUI* (Figure 6.6). The *GUI* autonomously acquires the calibration data with little human intervention. It is interfaced with a microcontroller through an emulated RS-232 connectivity over a physical USB. The microcontroller



hosts the firmware that is needed to protocolize the interface and as well as to control two stepper motors. The camera to be calibrated is also connected to the computer. Manufacturer provided API is used to develop the necessary software to control the camera. The computer stores the image data and processes the data offline once the data acquisition is completed. In the following sections the calibration rig and the spherical calibration target is discussed in details.

### 6.2.1 The Calibration Rig

The calibration rig (Figure 6.7) is a rigid structure with a mechanized calibration table with two degrees of freedom of translatory motion. It houses the camera and the calibration table and provides a rigid framework for the task of point correspondence generation. The camera is mounted on the calibration rig to ensure rigidity during the calibration procedure. In addition, the calibration target is mounted on the calibration table (Figure 6.8) which is displaced accurately by two orthogonal stepper motor driven ball screws in a plane through a rectangular calibration grid. The basic resolution of the stepper motor ball screw assembly is  $\frac{1}{2000}$  inch which allows the world coordinates of the calibration points to be determined with high accuracy.

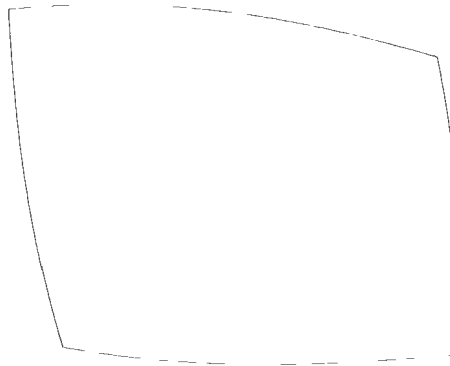


Figure 6.3: Perspective Distortion in the Synthesized Image of a Square with Strong Lens Distortion Effect

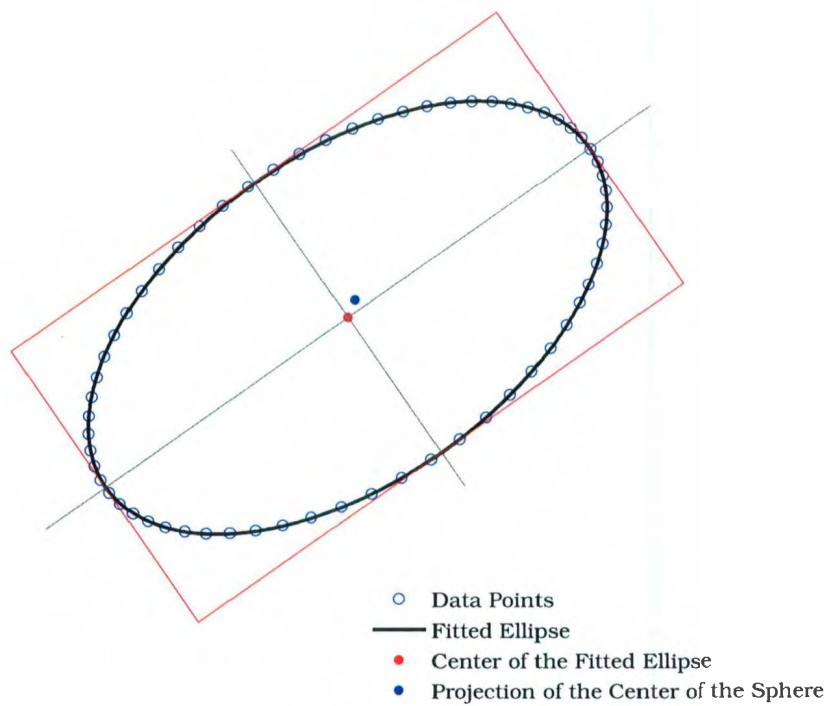


Figure 6.4: Perspective Distortion in the Synthesized Image of a Circle with Strong Lens Distortion Effect

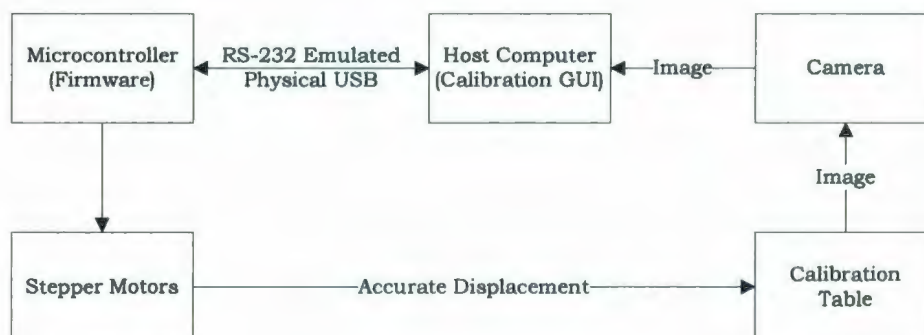


Figure 6.5: Calibration System

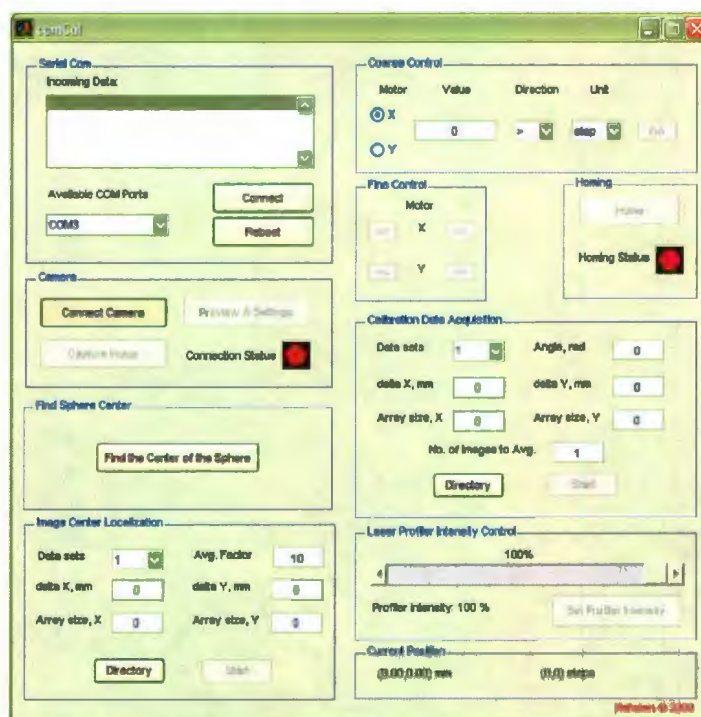


Figure 6.6: GUI for Calibration Data Acquisition

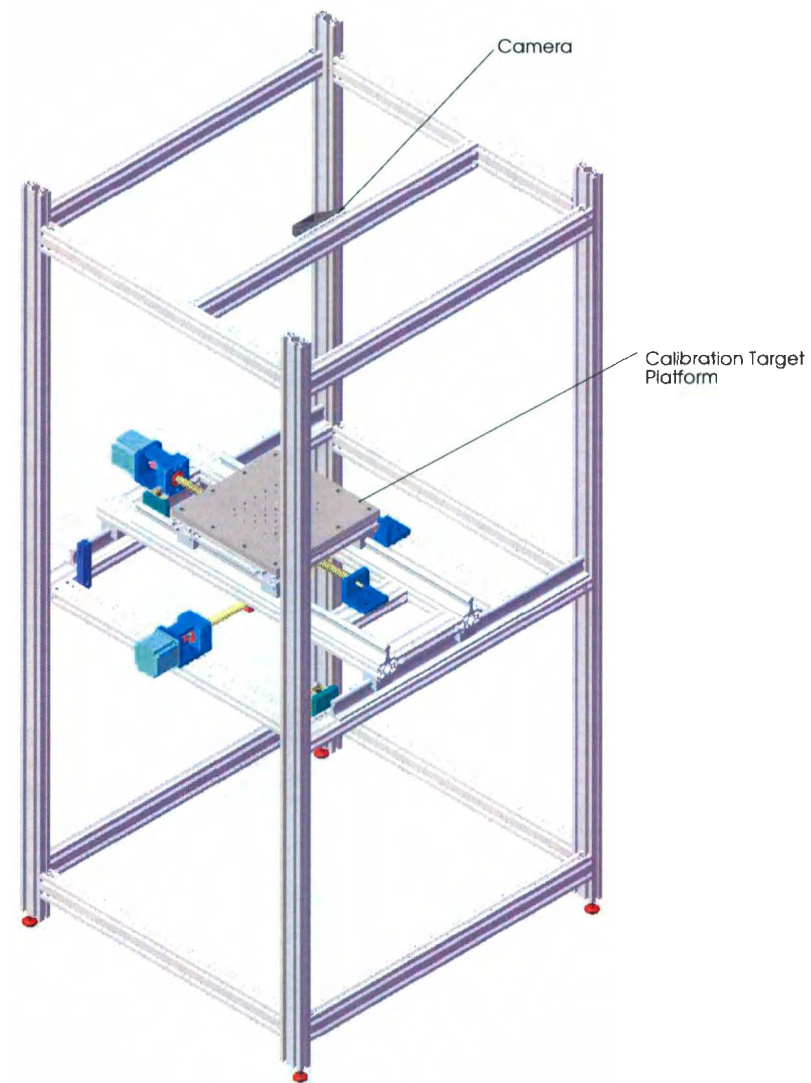


Figure 6.7: Calibration Rig



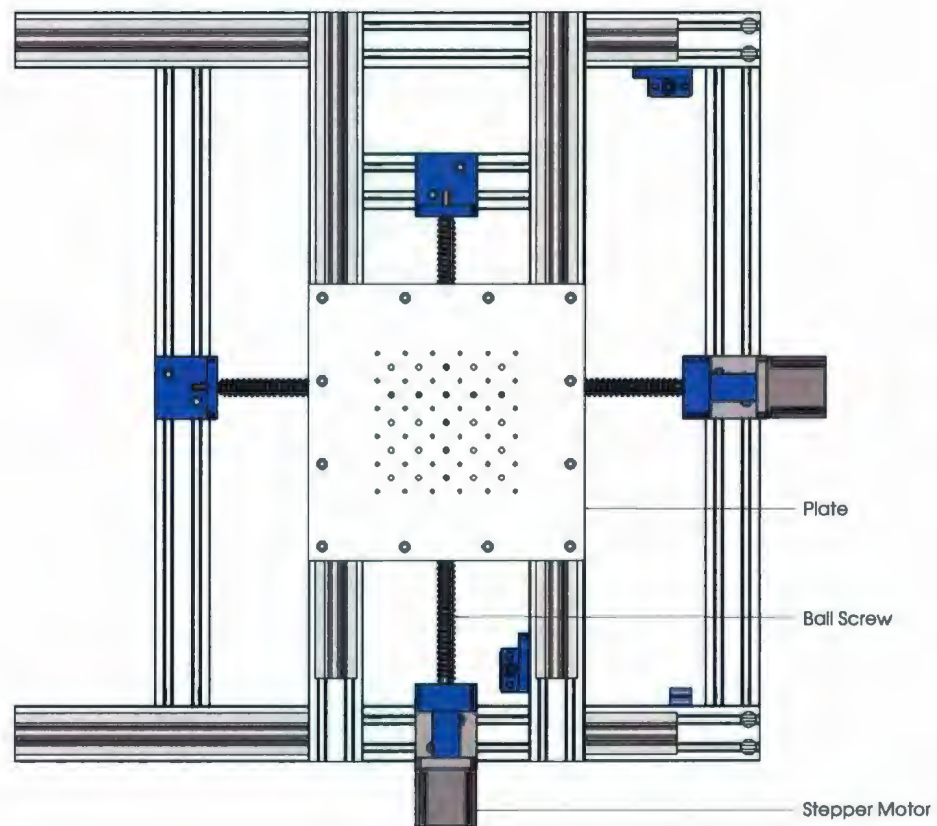


Figure 6.8: Calibration Table

### 6.2.2 Spherical Calibration Target

A precisely machined ceramic sphere is used as the calibration target (Figure 6.10). The center of the sphere represents the calibration points. A calibration grid is defined to determine the positions of the calibration points. The stepper motor driven ball screws displace the calibration target mounted on the calibration table through the nodes of the calibration grid. At each node the camera acquires an image of the calibration target and stores it in the host computer for future processing. Suitable image feature extraction technique localizes the geometric center of the projective image of the occluded contour of the sphere. Primarily this geometric center represents the image of the respective calibration point. Additional mechanical fixtures elevate the calibration target to different heights to generate a multi planar array of point correspondences.

There are some distinct advantages that the spherical calibration target can offer over conventional targets:

- The occluded contour of the sphere in the perspective projection is always a circle irrespective of the position of the target and is imaged as an ellipse. Localizing the contour in the image space is comparatively simple and can be done with greater accuracy. Perspective distortion is also minimized in this approach. It is also practically uniform over the field of view of the camera.
- Unlike the dot pattern or the checkerboard pattern, the sphere can occupy a larger area than an image of a single dot or square. Hence, greater number of data points can be acquired to localize the image of the sphere with higher accuracy.

- Since a single calibration target is displaced in the field of view of the camera, the calibration grid can be designed to populate greater number of point correspondences. The grid spacing can be chosen arbitrarily since the image of the target at the neighboring nodes can overlap. On the other hand, the conventional target can not offer this kind of flexibility.
- The geometric center of the sphere in the image space is considered as the image of the calibration point which does not coincide the projection of the true center of the sphere in actuality. However, the unique geometry of the imaging process of a sphere (Figure 6.11) offers an opportunity to compensate for this discrepancy mathematically. This is not achievable as accurately in case of a conventional target.
- The radiometric properties of a sphere under controlled illumination is more favorable than the conventional target as far as the image feature extraction technique is concerned (Figure 6.9).
- Since the calibration data is generated from greater number of images, the data inherently minimizes the effect of the random measurement error introduced by the camera (Section 6.4).



Figure 6.9: Radiometric Properties of a Spherical Calibration Target and a Circular Calibration Target

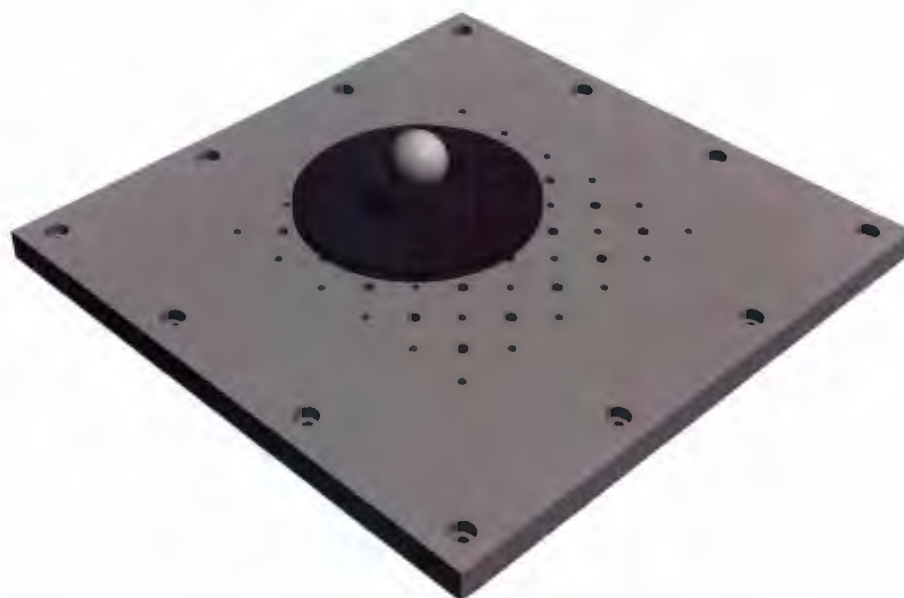


Figure 6.10: Spherical Calibration Target Mounted on the Calibration Table  
(Sphere Diameter = 25.4 mm)



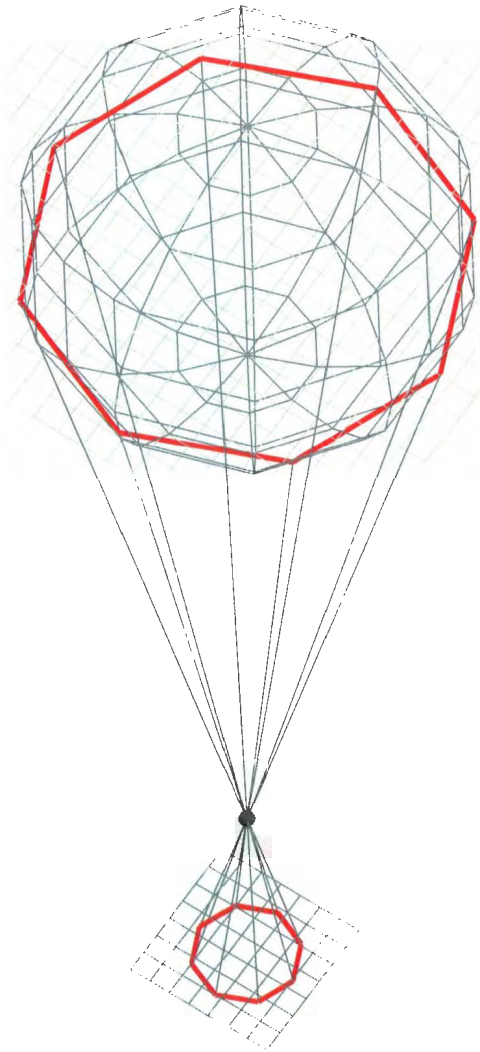


Figure 6.11: Perspective Projection of a Sphere (*3D* Model)

## 6.3 Imaging a Sphere

In this section the imaging process of a sphere is modeled mathematically. Figure 6.11 presents a 3D model of the perspective projection of a sphere. The image of the sphere can be produced by drawing tangents on the sphere through the center of projection  $C$  (i.e., optical center of the lens). The points at which these tangents intersect the image plane provide the image of the occluded contour of the sphere.

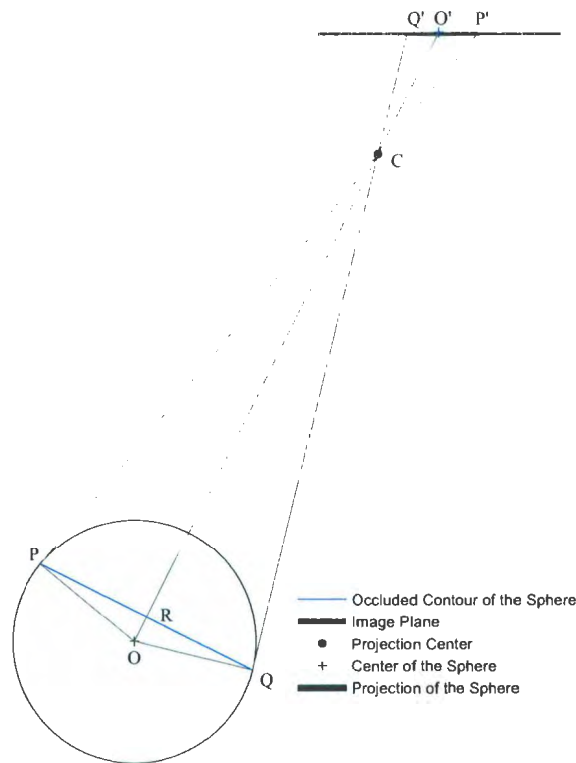


Figure 6.12: Perspective Projection of a Sphere (Cross-section through a General Plane Containing the Center of the Sphere and the center of projection)

Figure 6.12 presents a cross-section of the geometrical model (Figure 6.11) through a general plane that contains the center of the sphere and the center of projection  $C$ . This effectively reduces the model from 3D to 2D.  $CP$  and  $CQ$  are the tangent lines on the circle representing the sphere, drawn from the center of projection. It can be proven from basic geometrical definitions that,  $CP = CQ = \ell$ . Since  $OQ \perp CQ$  and  $OP \perp CP$ ,  $\triangle OPC$  and  $\triangle OQC$  are both right angled triangles. Letting  $OC = d$  and  $OP = OQ = r$  and applying the Pythagorean theorem on either of the right angled triangles yields the following expression for  $\ell$ :

$$\ell = \sqrt{d^2 - r^2} \quad (6.1)$$

It can be easily shown that the triangles  $\triangle OQC$  and  $\triangle OPC$  are congruent; i.e.,  $\triangle OQC \cong \triangle OPC$ . Therefore,  $\angle POC = \angle QOC = \theta$ . Subsequently, congruency of the two triangles  $\triangle OPR$  and  $\triangle OQR$  can be proven from  $\angle POC = \angle QOC = \theta$ . This eventually leads to the conclusion that  $OC \perp PQ$ .  $OC$  and  $PQ$  intersect each other at point  $R$ . In order to find the position of the point  $R$ ,  $CR = d_r$  is defined in the following equation:

$$\begin{aligned} d_r &= \ell \cos \theta \\ &= \frac{\ell^2}{d} \quad \left[ \because \cos \theta = \frac{\ell}{d} \right] \end{aligned} \quad (6.2)$$

The points  $P$  and  $Q$  can be found by intersecting the circle centered at point  $O$  with another circle centered at the projection point  $C$  with radius  $\ell$ . If the model is extended to 3D from 2D, the locus of the points  $P$  or  $Q$  in 3D will be the intersection points of the spherical calibration target centered at point  $O$  and another sphere centered at the projection point  $C$  with radius  $\ell$ . Since  $\ell + r > d$ , the intersection points will constitute a circle  $I$  centered at point  $R$  with radius  $PR = QR = r_0$ . The

length of the radius  $r_0$  can be determined according to the following equation:

$$\begin{aligned} r_0 &= \ell \sin \theta \\ &= \frac{\ell r}{d} \quad \left[ \because \cos \theta = \frac{r}{d} \right] \end{aligned} \quad (6.3)$$

In order to locate this intersection circle  $I$  with respect to the camera coordinate frame  $\mathbf{C}$  (originated at the center of projection  $C$ ), another coordinate frame  $\mathbf{C}'$  is defined whose  $Z$  axis is along the vector  $\vec{CO}$ .  $X$  and  $Y$  axes of the coordinate frame  $\mathbf{C}'$  lies on the plane that contains the intersection circle  $I$ . The direction of either of the two axes  $X$  and  $Y$  can be chosen arbitrarily. If the direction of  $X$  axis is chosen arbitrarily, subsequently  $Y$  axis can be found by the cross product of the axes  $Z$  and  $X$ ; i.e.,  $Y = Z \times X$ . Let the unit vectors along the axes of the coordinate frame  $\mathbf{C}'$  be  $\hat{u}, \hat{v}, \hat{w}$  respectively. The general expression for the position vector of the intersection point  $P$  or  $Q$  with respect to the coordinate frame  $\mathbf{C}'$  is provided by Equation 6.4.

$$\begin{aligned} \vec{RP} &= r_0 \cos t \hat{u} + r_0 \sin t \hat{v} + 0 \hat{w} \quad \left[ \because \text{Point } P \text{ lies in the } X - Y \text{ plane of } \mathbf{C}' \right] \\ &= r_0 \cos t \hat{u} + r_0 \sin t \hat{v} \end{aligned} \quad (6.4)$$

In Equation 6.4,  $t$  is an independent parameter defined within the real interval  $[-\pi, \pi]$ . The position vector of the point  $P$  with respect to the camera coordinate frame  $\mathbf{C}$  is calculated in accordance with Figure 6.12 (Equation 6.5).

$$\vec{CP} = \vec{CR} + \vec{RP} \quad (6.5)$$

Let  $\hat{i}, \hat{j}$  and  $\hat{k}$  be the orthogonal unit vectors along  $X, Y$  and  $Z$  axes of the coordinate frame  $\mathbf{C}$  respectively. The orthogonal unit vectors  $\hat{u}, \hat{v}$  and  $\hat{w}$  of the coordinate frame



$C'$  can be expressed with respect to the camera coordinate frame  $C'$ :

$$\hat{u} = u_x \hat{i} + u_y \hat{j} + u_z \hat{k} \quad (6.6)$$

$$\hat{v} = v_x \hat{i} + v_y \hat{j} + v_z \hat{k} \quad (6.7)$$

$$\hat{w} = w_x \hat{i} + w_y \hat{j} + w_z \hat{k} \quad (6.8)$$

Please note that,  $\vec{C}R = d_r(w_x \hat{i} + w_y \hat{j} + w_z \hat{k})$ . Expressing the vectors in Equation 6.5 with respect to the camera coordinate frame  $C$  yields the position vector of point  $P$  with respect to the same coordinate frame (Equation 6.9).

$$\begin{aligned} \vec{C}P &= d_r(w_x \hat{i} + w_y \hat{j} + w_z \hat{k}) + (r_0 \cos t \hat{u} + r_0 \sin t \hat{v}) \\ &= d_r(w_x \hat{i} + w_y \hat{j} + w_z \hat{k}) + r_0 \cos t (u_x \hat{i} + u_y \hat{j} + u_z \hat{k}) + r_0 \sin t (v_x \hat{i} + v_y \hat{j} + v_z \hat{k}) \\ &= \begin{bmatrix} u_x & v_x & w_x \\ u_y & v_y & w_y \\ u_z & v_z & w_z \end{bmatrix} \times \begin{bmatrix} r_0 \cos t \\ r_0 \sin t \\ d_r \end{bmatrix} \end{aligned} \quad (6.9)$$

Equation 6.9 provides the general position vector for all the points on the intersection circle  $I$ . Projective transformation of this position vector will produce the undistorted image of the occluded contour of the sphere. Subsequently addition of lens distortion components will yield the distorted image of the sphere.

## 6.4 Measurement Error of the Camera

The digital camera is incapable of producing identical images of the scene in multiple exposures even with controlled ambient illumination. This can be attributed to photon noise and the internal electronics associated with the discretization of the

CCD signal. In this section this phenomenon is studied with an aim to quantify the accuracy associated with the localization of the geometric center of the image of the sphere.

Two sets of one thousand images at the same position are sampled by the camera for two different spherical calibration targets of the same radius but made of two different materials; namely, delrin and ceramic. It should be noted that according to the vendor the ceramic sphere is relatively more accurate in terms of sphericity due to its sophisticated manufacturing process. The image feature extraction technique retrieves the geometrical center of the sphere in each image. The distributions of the centers are presented in Figure 6.13. Since the histograms of the  $X$  and  $Y$  coordinates of the geometric centers in Figure 6.14 exhibit normally distributed data, the measurement error can be considered to be a random occurrence. The statistical measures of the distributions are presented in Table 6.1. In order to minimize the presence of the random error, it is preferable to acquire multiple images at each node of the calibration grid and subsequently averaging them to a single image. The center localized in the average image provides a better estimate of the true geometric center of the image of the sphere.

It is evident from the above discussion that the measurement error introduced by the camera can appreciably contribute to the noise present in the calibration data. The noise becomes more prominent when only a few images of the calibration target are analyzed to generate the point correspondences. Therefore, it is highly recommended that multiple images of the calibration target at each grid node be acquired to attain higher accuracy in the calibration data.

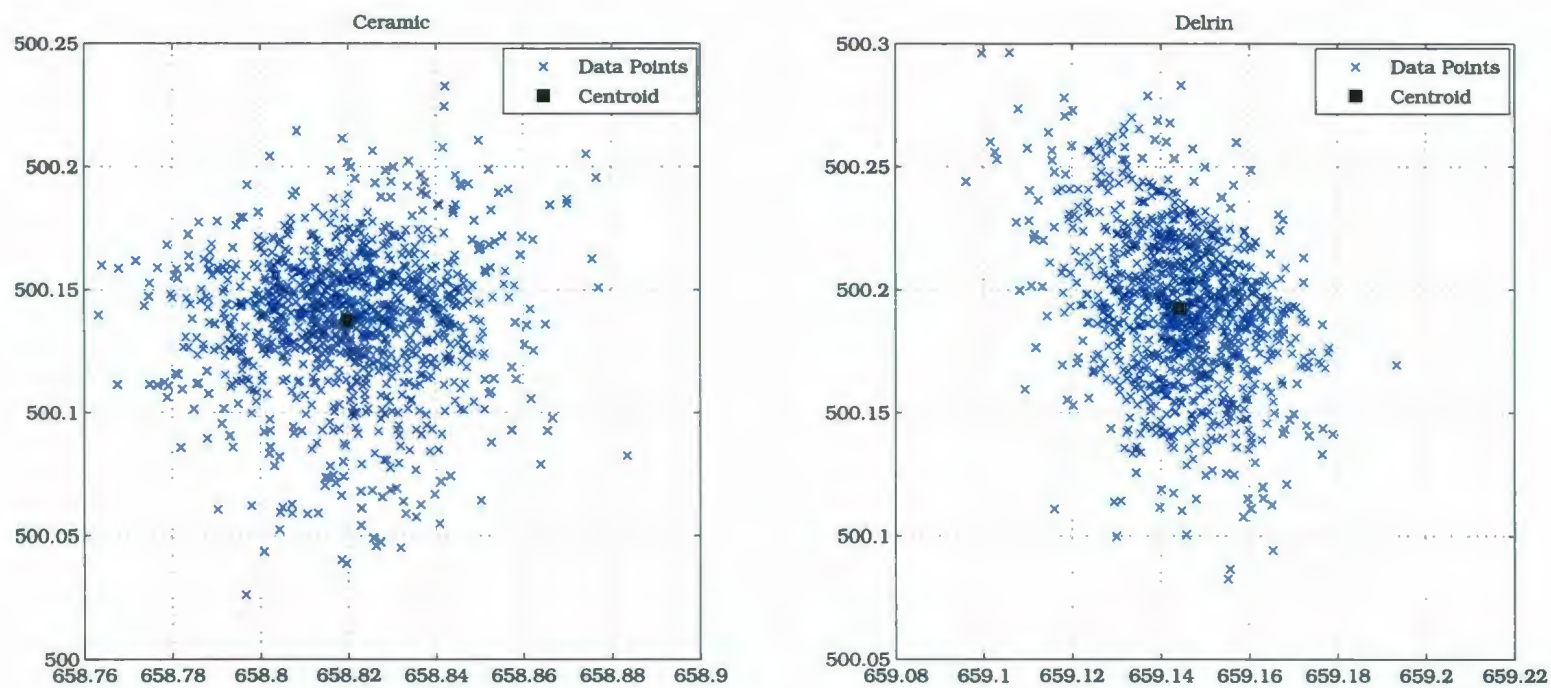


Figure 6.13: Distribution of the Geometric Centers of the Images of the Spheres (Ceramic & Delrin)

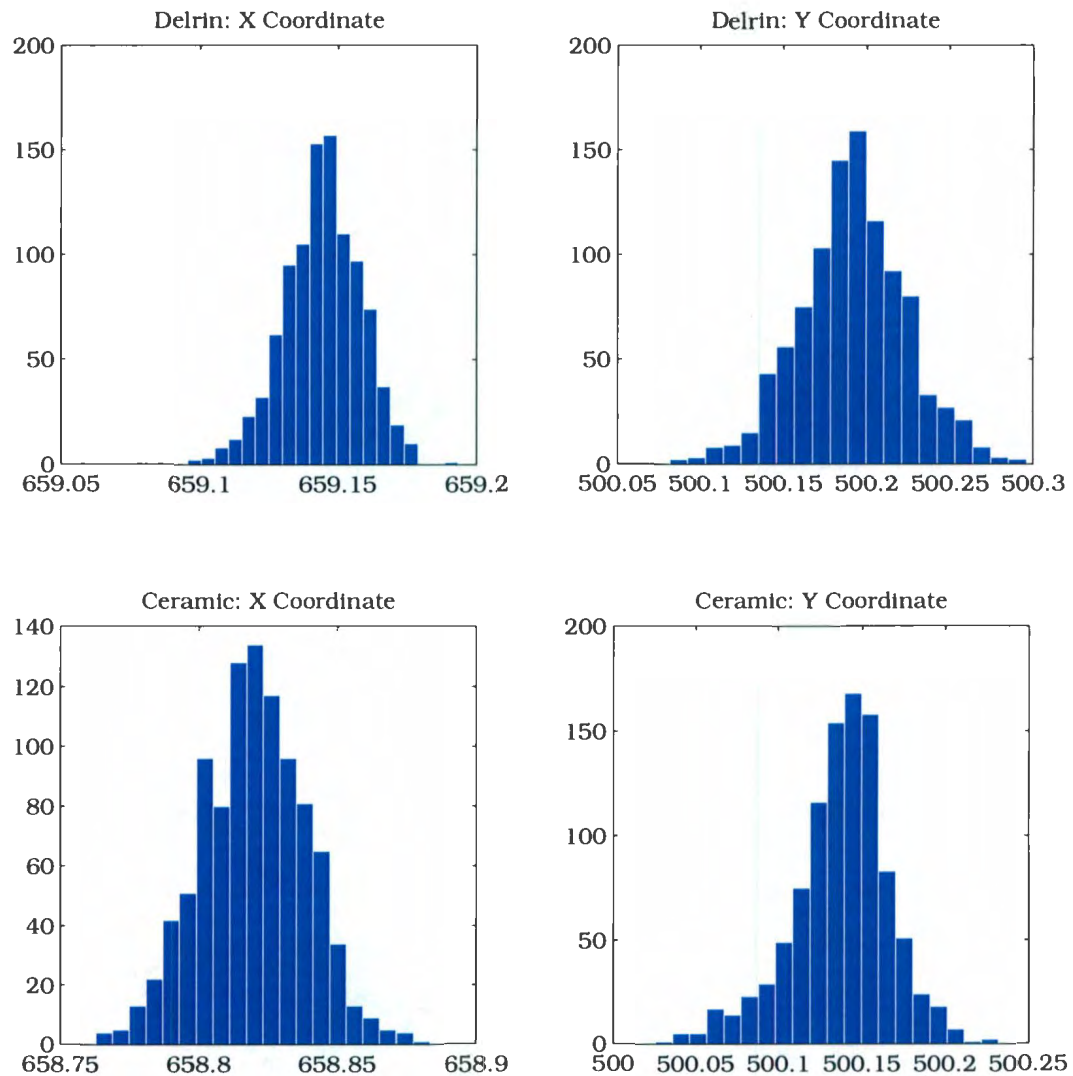


Figure 6.14: Histogram of X and Y Coordinates of the Geometric Centers of the Images of the Spheres (Ceramic & Delrin)



Table 6.1: Statistical Measures of the Distributions of Center Locations in the  
Image Space

	$\mu$	$\sigma$	95% C.I.
Delrin ( $X$ )	659.1445	0.0137	659.1445 $\pm$ 0.0009
Ceramic ( $X$ )	658.8199	0.0189	658.8199 $\pm$ 0.0012
Delrin ( $Y$ )	500.1925	0.0319	500.1925 $\pm$ 0.0020
Ceramic ( $Y$ )	500.1375	0.0295	500.1375 $\pm$ 0.0018

## 6.5 Localization of the Calibration Points in the Image Space

A customized image feature extraction technique was developed to localize the occluded contour of the spherical target in the image. The contour thus localized was fitted to an ellipse by least squares analysis. The geometric center of the ellipse was approximated as the image of the calibration point. The image extraction technique is presented in Figure 6.15.

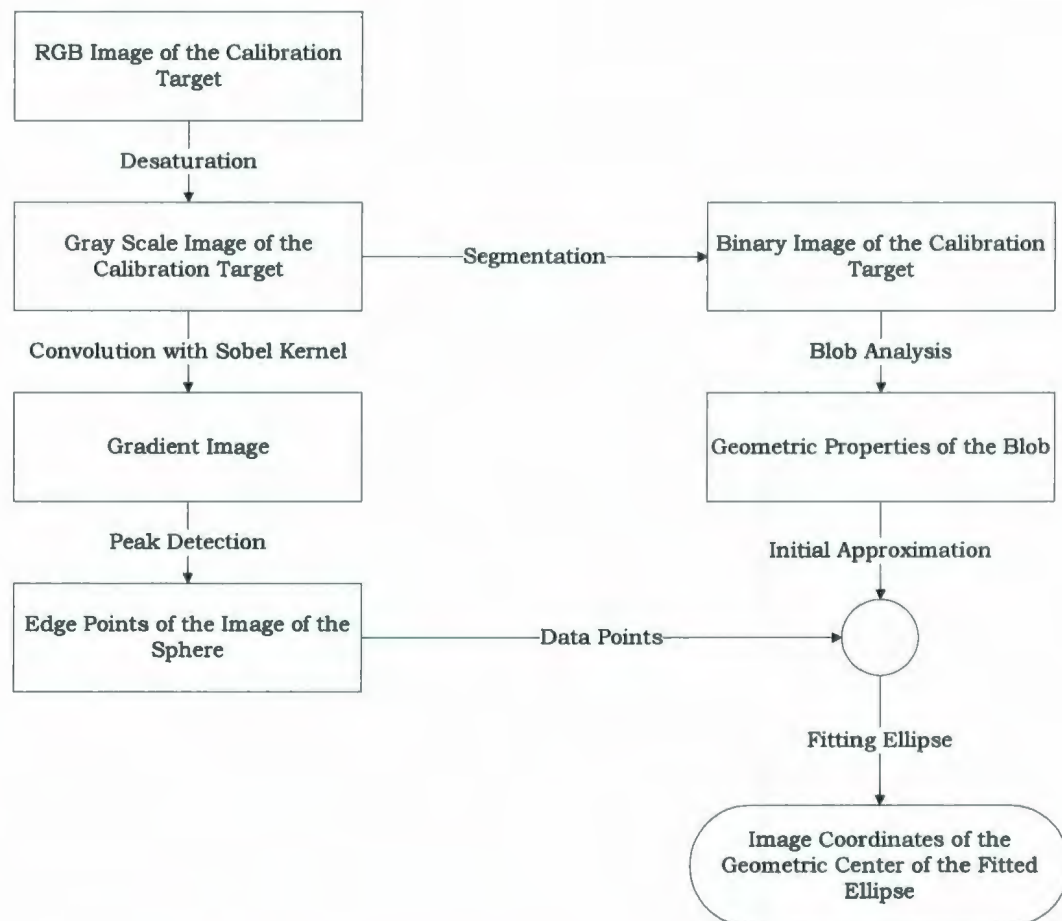


Figure 6.15: Image Feature Extraction Technique

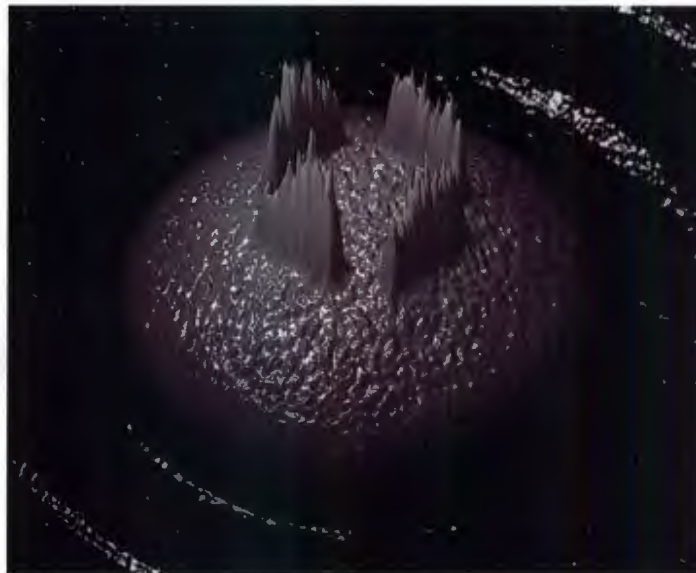


Figure 6.16: 3D Plot of the Gray Scale Image (Gray Values are Plotted in 3D Against Their Respective Pixel Positions)

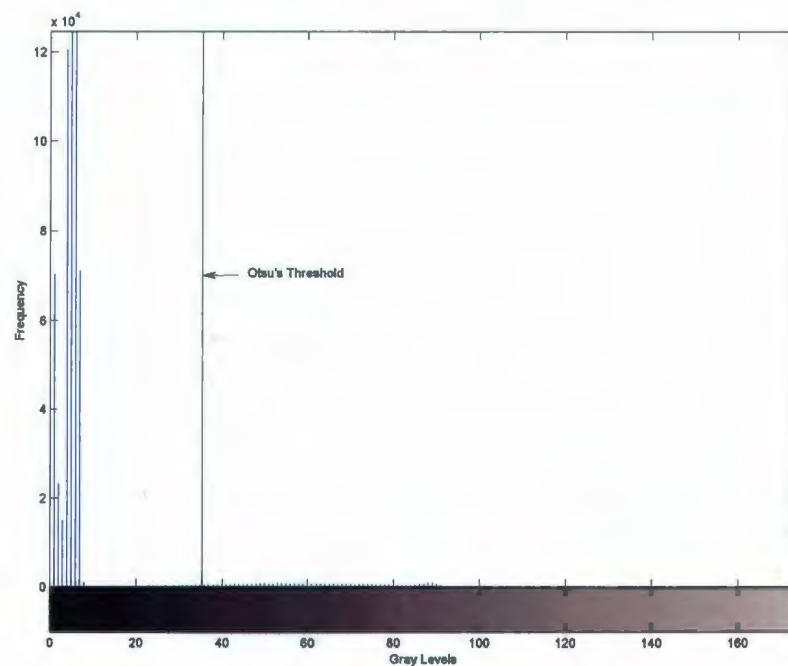


Figure 6.17: Histogram of a Generic Gray Scale Image of the Calibration Target



Figure 6.18: Gradient Image with the Peaks Localized

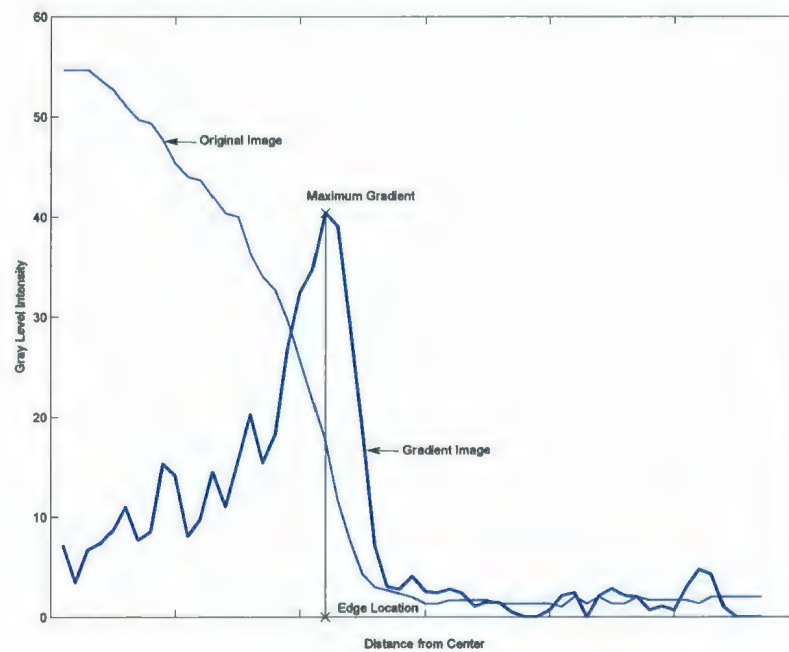


Figure 6.19: Gray Scale Intensity and the Corresponding First Derivative of the Spatial Signal in the Radial Direction



The RGB image of the calibration target can be desaturated by averaging all the three color channels to convert it to a gray scale image (Figure 6.16). Subsequently segmenting the gray scale image by Otsu's method [40] (Figure 6.17) yields the binary image of the calibration target (Figure 6.20).

Since occurrence of the random noise is inevitable, the binary image contains a number of noisy blobs; however, the largest blob always represents the image of the calibration target. Blob analysis of the binary image provides the geometric properties of the image of the spherical target. These properties include the centroid of the blob, the principal axes of inertia, location and dimensions of the rectangle bounding the blob with minimum area (i.e., minimum bounding rectangle). All these parameters provide a good initial approximation for fitting the ellipse by nonlinear least squares analysis.



Figure 6.20: Segmented Image (Localized Contour, Minimum Bounding Rectangle, Major Axis of Inertia)

In order to localize the edge points of the occluded contour with sub pixel accuracy, the gray cycle image is convolved with a Sobel Kernel to yield the corresponding gradient image (Figure 6.18). Since the direction of the edge is radial at all points, an exhaustive search is executed to detect the peaks in the radial direction that represent the edge points of the contour of the calibration target in the image (Figure 6.19). Subsequently these edge points are fitted to an ellipse by least squares analysis. Since the fitting of an ellipse by least squares analysis is crucial in terms of sub pixel localization of the edge and consequently the geometric center, in the following section a detailed description is presented.

## 6.6 Fitting Ellipse by Least Squares Analysis

Fitting geometric primitives (e.g., lines, circles, ellipses etc.) to a given set of data points is one of the classic problems encountered in computer graphics. Since the projective image of a sphere or a circle is an ellipse, fitting an ellipse to the edge points located by the image analysis will effectively localize the projection of the sphere or the circle in the image space. This problem is attempted in two different approaches in the literature; however, either the linear or the nonlinear least squares analysis is adopted as the general method of solution. These two approaches are generally termed as the algebraic approach and the geometric approach.

### 6.6.1 Algebraic Fit

In the algebraic approach, the data points are fitted to the general conic section equation (Equation 6.10) by least squares analysis.

$$ax^2 + bxy + cy^2 + dx + ey + f = 0 \quad \text{with, } a^2 + b^2 + c^2 \neq 0 \quad (6.10)$$

The quadratic in Equation 6.10 encompasses all the four types of conic sections; namely, ellipse, circle, parabola and hyperbola. If  $b^2 - 4ac < 0$ , Equation 6.10 represents an ellipse; hence, algebraically fitting an ellipse to a number of data points essentially becomes a constrained least squares analysis problem with the constraint being  $b^2 - 4ac < 0$ . References ([41, 42, 43, 44]) adopted this algebraic approach for ellipse fitting; however, these references mainly differ in constraining the quadratic equation (Equation 6.10) in different unique forms.

### 6.6.2 Geometric Fit

The geometric distances between the given data points and the fitted ellipse are minimized in the geometric approach. This approach fits an ellipse to the general parametric form of the equation of an ellipse. Essentially this approach is iterative; therefore, it is computation intensive. While the algebraic approach is direct and computationally inexpensive, the geometric approach yields better accuracy at the cost of greater computational effort. From the standpoint of accurately calibrating the camera, adopting this approach in this work is therefore well justified. In [45], a detailed comparative account on fitting ellipse by least squares analysis is offered highlighting both the geometric and the algebraic approach.



## 6.7 Geometric Fit of Ellipse in Parametric Form

The parametric equation of an ellipse in the canonical position (i.e., the ellipse is centered at the coordinate origin and the axes coincide with the coordinate axes) is provided by:

$$\mathbf{E} = \begin{bmatrix} a \cos \psi \\ b \sin \psi \end{bmatrix} \quad (6.11)$$

In Equation 6.11 the parameter  $\psi$  is the *eccentric anomaly* independently defined within the interval  $[0, 2\pi]$ . At any point  $(x, y)$  on the ellipse, the expression for  $\psi$  is:

$$\psi = \arctan \frac{ay}{bx}$$

The general ellipse is unconstrained; i.e., it is centered at any arbitrary point and the axes are also oriented arbitrarily. Let the center of the ellipse in Equation 6.11 be translated to an arbitrary point  $(t_x, t_y)$ . It is subsequently rotated such that the major axis of the ellipse produces an angle  $\theta$  with the coordinate  $X$  axis. The transformation of the canonical ellipse by the aforementioned translation and subsequent rotation provides the unconstrained general ellipse (Equation 6.12).

$$\begin{aligned} \mathbf{E} &= \begin{bmatrix} t_x \\ t_y \end{bmatrix} + R(\theta) \times \begin{bmatrix} a \cos \psi \\ b \sin \psi \end{bmatrix} \\ &= \begin{bmatrix} t_x + a \cos \psi \cos \theta - b \sin \psi \sin \theta \\ t_y + a \cos \psi \sin \theta + b \sin \psi \cos \theta \end{bmatrix} \end{aligned} \quad (6.12)$$

In Equation 6.12  $R(\theta)$  is the 2D rotation matrix for counterclockwise rotation.

$$R(\theta) = \begin{bmatrix} \cos \theta & -\sin \theta \\ \sin \theta & \cos \theta \end{bmatrix} \quad (6.13)$$



Given a set of  $n$  data points  $S = \{(x_1, y_1), (x_2, y_2), (x_3, y_3) \dots (x_n, y_n)\}$  the general problem is to find an ellipse  $E$  that minimizes the geometric distances between the data points and the fitted ellipse. The parameters that define the ellipse  $E$  completely are the position of the center  $(t_x, t_y)$ , half length of the axes  $a$  and  $b$  and the angle between the major axis and the coordinate  $X$  axis  $\theta$ . If  $n > 5$ , these parameters can be estimated by the Newton-Gauss nonlinear least squares analysis.

Since the eccentric anomaly  $\psi$  at the given data points can't be determined because of the initial unavailability of the location of the center and the orientation of the axes, for each data point  $\psi$  is considered to be an additional unknown. Hence, for  $n$  given data points the unknown parameter vector to be estimated is  $[\psi_1 \ \psi_2 \ \psi_3 \dots \psi_n \ t_x \ t_y \ a \ b \ \theta]^T$ .

An initial approximation of the ellipse parameters is required to initiate the iterative search for better estimates. The geometric properties of the blob representing the sphere in the image can provide good approximations of the ellipse parameters. The centroid of the blob approximates the center of the ellipse. In addition, the orientation and the length and width of the minimum bounding rectangle provide good approximations for the parameters  $a$ ,  $b$  and  $\theta$ .

The Jacobian matrix for the least squares problem becomes singular when  $b = a$ . If initial approximations of  $b$  and  $a$  are very close ( $a \approx b$ ), this singularity can be avoided by arbitrarily setting  $b = \frac{4}{5}a$ . However, the analysis still may fail to converge to a solution when the surface of the occluded contour of the sphere is practically parallel to the image plane resulting in a circular image. The unavailability of a solution is duly handled in the image analysis by fitting a circle to the edge points. Figure 6.21 presents an ellipse estimated by the aforementioned analysis.

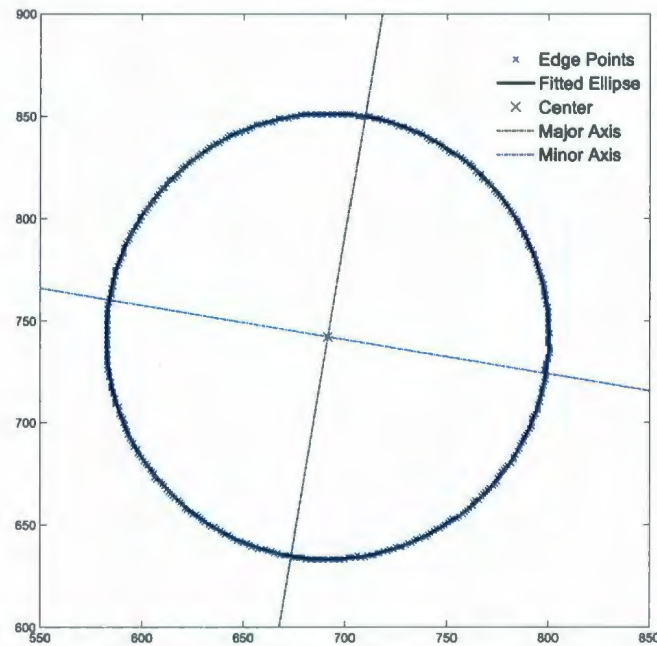


Figure 6.21: Fitting Ellipse to the Edge Points

## 6.8 Practicalities in Camera Calibration

The nodes of a multiplanar rectangular calibration grid were considered to represent the position of each calibration point in the object space. The calibration system positioned the calibration target (Figure 6.10) at the nodes. The camera acquires multiple images at each position and averaged them into a single image to minimize the measurement error of the camera. The averaged images were stored in the host computer for off line processing. Once all the images were acquired, the image feature extraction technique approximated the locations of the calibration point in the image space as the geometrical centers of the contour of the calibration target. The calibration data thus produced were used to obtain the unrefined camera model. Since in a strict sense the geometrical center and the projection of the center of the sphere do



Figure 6.22: Superimposed Fitted Ellipse on Actual Calibration Image

not coincide, subsequent refinement is required to compensate for this discrepancy.

The ellipse fitted by the least squares analysis represents the image of the occluded contour of the calibration target. The optical center of the lens and the fitted ellipse constitute an oblique cone whose axis represents the projection ray of the center of the spherical target. The point at which the projection ray intersects the image plane is the actual image point of the center of the sphere. Using the unrefined camera model, these image points can be estimated to refine the calibration data. The modified data is then calibrated to obtain the refined camera model.

## Chapter 7

# Performance Evaluation of the Proposed Camera Calibration Technique

Benchmarking can be defined as the process of comparing quantitative entities against a similar reference point or standard of excellence [46]. Since accuracy is regarded as the primary performance index for a camera calibration technique, benchmarking the performance of the proposed technique is essentially a comparative study on accuracy attained against the accuracy provided by a standard technique. The term “accuracy” has two different aspects when it is referred to a camera model:

- The accuracy of the estimates of the camera parameters compared against their true values.
- The accuracy of the camera model in terms of minimizing the disagreement between the practical calibration data and reconstructed calibration data.



This chapter visits both of these two aspects of accuracy of a calibrated camera model.

In practice it is difficult, if not impossible, to physically measure the camera parameters. Even if such measurement technique existed which is capable of measuring the camera parameters physically, the presence of random error in the measurements can never be entirely eliminated. In consequence, the true values of the camera parameters in a strict sense can never be known. This nullifies the idea of assessing the accuracy of the estimates of the camera parameters against the respective true values. However, accuracy assessment of this nature is still possible by computer simulation where a reasonable analytic camera model is employed to synthesize images of a virtual calibration target. Thus point correspondences can be generated between the object space and the image space which in turn, can be used as an input to the camera calibration technique. The comparison of the calibrated camera model against the analytic camera model provides an index for the performance of the respective camera calibration technique. Since the presence of random error can not be helped in physical measurements, in the simulation the occurrence of the random error is treated as a stochastic process under Monte-Carlo method.

Unlike computer simulation, in a practical calibration scenario the true values of the camera parameters are usually not available. Hence, performance evaluation of a camera calibration technique in practice is based on  $2D$  or  $3D$  measurements. The calibrated camera model is used for vision based measurements of some geometric quantity. The comparison between the vision based measurements and the conventional physical measurements provides an index for performance. The accuracy attained by the proposed technique using practical calibration data is compared against that provided by a standard technique to assess the performance of the pro-

posed technique (Section 7.2).

Performance evaluation always involves a predefined standard against which the comparison is made. In this study, the camera calibration technique by Heikkila [4] is chosen as the reference calibration technique, since it is recognized as one of the prominent techniques for camera calibration [11]. Moreover, a MATLAB<sup>®</sup> based implementation produced by Heikkila himself is available online. It is particularly convenient since Heikkila's code can be seamlessly integrated with this work. Therefore, this technique is used for the simulation and as well as for practical calibration. The code is available online at <http://www.ee.oulu.fi/~jth/calibr/>.

## 7.1 Performance Evaluation by Monte-Carlo Simulation

Any real measurement involves errors that can be divided into four classes [7]:

- Blunders
- Constant Errors
- Systematic Errors
- Random Errors

Blunders and constant errors are irrelevant in the context of computer simulation. On the other hand, known systematic error in a vision based measurement system is caused by lens distortion, which is duly compensated for in the camera model.

Occurrence of random errors can be attributed to inherent incapability of the instruments, but their occurrence can be modeled in the computer simulation by employing pseudo-random numbers.

A Monte Carlo simulation is defined as the method of calculating some physical quantity that involves deliberate use of random numbers that has the structure of a stochastic process [47]. In synthesizing the image of a virtual calibration target by an analytic camera model, pseudo-random numbers are used to introduce random error in the imaging process. Hence, this computer simulation can be treated as a Monte-Carlo simulation process.

### 7.1.1 Synthesizing the Image of a Virtual Calibration Target

The following steps are generally followed to generate the synthetic image (i.e., calibration data) of a virtual calibration target:

- **Step 1:** Assume an analytic camera model. The camera model should fully define the image formation process. Therefore, reasonable values for all the camera parameters that are listed in Table 2.1 must be chosen.
- **Step 2:** Choose a virtual calibration target. The range and the geometry of the target should be such that the image of the target covers the whole field of view of the analytic camera. The target should contain an array of calibration points. With respect to some world coordinate frame the coordinates  $[x_w y_w z_w]^T$  of all the calibration points are known.
- **Step 3:** Introduce random noise  $\epsilon_r$  to the world coordinates of the calibration points to generate the noisy world coordinates  $[x'_w y'_w z'_w]^T$  (Equation 7.1) of the



calibration points. The probability distribution function of the random error is assumed to be uniform. This error is defined within an error bound  $[\epsilon_{b+} \epsilon_{b-}]$ . The pseudo-random number representing the random error is normalized over the predefined error bound to ensure that the random error is always within that range.

$$\begin{bmatrix} x'_w \\ y'_w \\ z'_w \end{bmatrix} = \begin{bmatrix} x_w \\ y_w \\ z_w \end{bmatrix} + \begin{bmatrix} \epsilon_{rx} \\ \epsilon_{ry} \\ \epsilon_{rz} \end{bmatrix} \quad (7.1)$$

- **Step 4:** Generate the distorted image of the calibration target as described in Section 2.4 from the noisy world coordinates  $[x'_w y'_w z'_w]^T$  of the calibration points. The distorted image coordinates of the calibration points  $[u_d v_d]^T$  are stored for future analysis.
- **Step 5:** Input the noise-free world coordinates  $[x_w y_w z_w]^T$  and the distorted image coordinates  $[u_d v_d]^T$  (provided by the analytic camera using the noisy world coordinates of the calibration points) to the calibration technique. The calibration process yields estimates of the camera parameters.
- **Step 6:** Repeat steps 3, 4 and 5 for any arbitrary number ( $n$ ) of times. A large value of  $n$  yields better statistical inferences from the simulated data. Each time the estimates of the camera parameters from Step 5 are stored in the simulated data matrix  $S$ .

The simulated data matrix  $S$  contains  $n \times 15$  elements where each element represents the estimate of the respective camera parameter. Two such simulated data matrices



are generated, each one for the proposed and Heikkila's camera calibration technique.

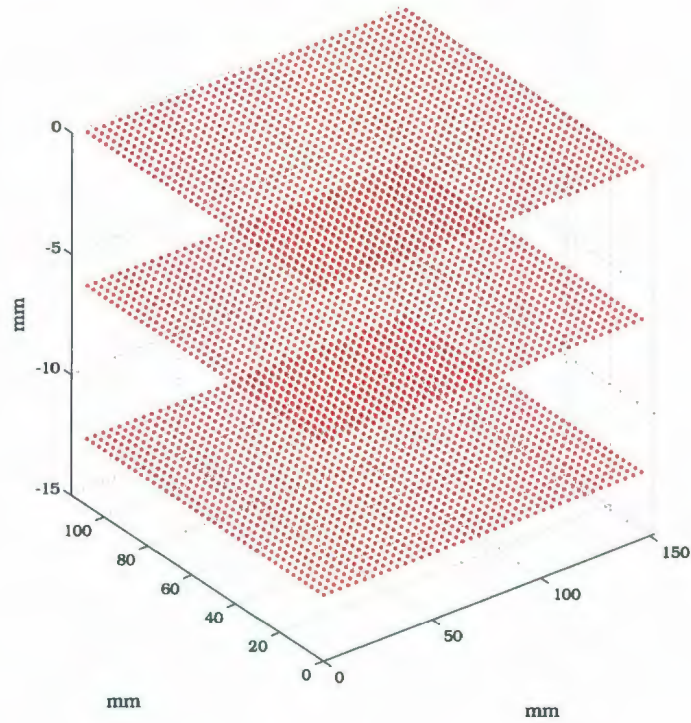


Figure 7.1: Virtual Calibration Target with  $40 \times 40 \times 3$  Calibration Points

### 7.1.2 Simulation Parameters

All the camera parameters except the lens distortion coefficients for the analytic camera are listed in Table 7.1. Two different lenses were considered, one with a weak lens distortion effect ( $L_W$ ) and the other with a strong lens distortion effect ( $L_S$ ). The values of the distortion coefficient for the lens  $L_S$  were obtained from [4]. On the other hand, the values of the distortion coefficients for the lens  $L_W$  were assumed

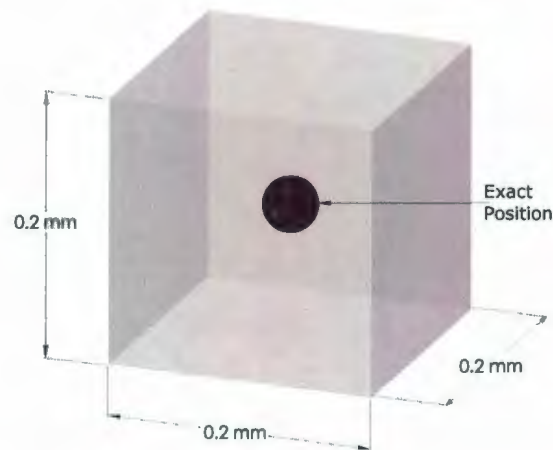


Figure 7.2: Defined Bound for Random Error

to be 10 times greater than that of the respective lens distortion coefficients of the lens  $L_S$ . These values are listed in Table 7.2. Additional simulation parameters are listed in Table 7.3.

### 7.1.3 Simulation Data Analysis

Each simulation run consists of 1000 trials where each trial provides different estimates of the camera parameters. Thus a distribution of estimates of each camera parameter is generated by each simulation run. The statistical means of these distributions approximate the analytic value of the respective camera parameter. To quantify the performance of the particular camera calibration technique in terms of accurately estimating the true values of the camera parameters, absolute percent error (*APE*) is used. *APE* is calculated according to the formula provided in Equation 7.2. In addition, the standard error of the mean (*SEM*) of the distributions provides an index

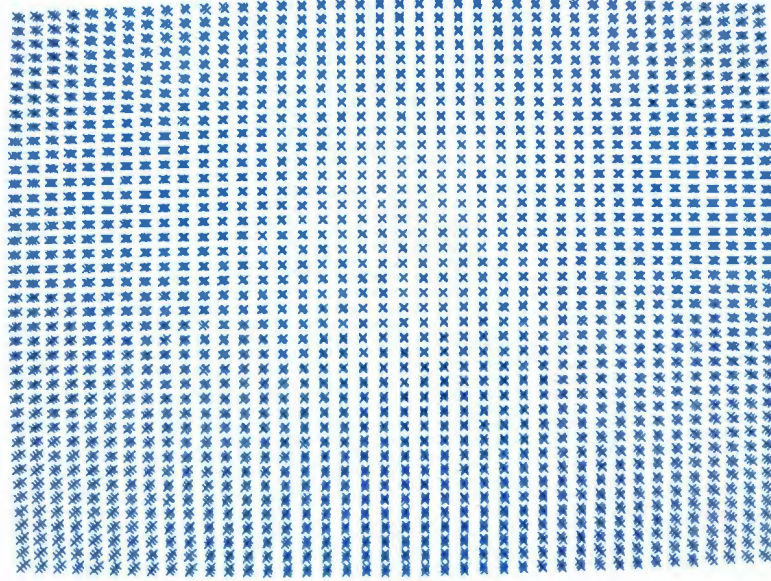


Figure 7.3: Image of the Virtual Calibration Target

of the sensitivity to noise for that particular camera parameter. In general, the higher is the value of  $SEM$ , the more the camera parameter is influenced by the random noise.  $SEM$  of the estimates of the camera parameters are calculated by the formula presented in Equation 7.3.

$$APE = \left| \frac{x - \bar{x}}{x} \right| \times 100\% \quad (7.2)$$

$$SEM = \frac{\sigma}{\sqrt{n}} \quad (7.3)$$

In Equation 7.2,  $x$  and  $\bar{x}$  refer to the analytical value and the estimated value of a camera parameter.  $\sigma$  and  $n$  in Equation 7.3 respectively represent the standard deviation of an estimate and the number of data points.



Table 7.1: Camera Parameters for the Simulation (Analytic Camera)

Camera Parameter	Symbol	Value	Unit
Scale factor	$s$	1.00	$n/a$
Effective focal length	$f$	8.50	$mm$
Image center	$(u_0, v_0)$	(650, 500)	$pixels$
Euler angles, XYZ	$\alpha$	$-1.5000^\circ$	$degrees$
	$\beta$	$+0.5000^\circ$	$degrees$
	$\gamma$	$+0.7500^\circ$	$degrees$
Equivalent quaternion (Appendix B)	$d$	+0.9999	$n/a$
	$a$	-0.0131	$n/a$
	$b$	+0.0043	$n/a$
	$c$	+0.0066	$n/a$
Position vector	$t_x$	-75.00	$mm$
	$t_y$	-55.00	$mm$
	$t_z$	+220.00	$mm$

Table 7.2: Lens Distortion Coefficients for the Analytic Lenses

	$L_W$	$L_S$	Unit
$k_1$	$3.30 \times 10^{-03}$	$3.30 \times 10^{-02}$	$mm^{-2}$
$k_2$	$-2.60 \times 10^{-05}$	$-2.60 \times 10^{-04}$	$mm^{-4}$
$p_1$	$-1.30 \times 10^{-06}$	$-1.30 \times 10^{-05}$	$mm^{-1}$
$p_2$	$4.00 \times 10^{-05}$	$4.00 \times 10^{-04}$	$mm^{-1}$



Table 7.3: Simulation Parameters

Parameter	Remarks
Calibration Target	3 parallels planes, each plane consisting uniformly spaced grid of calibration points over the rectangular area of $150.0 \times 110.5 \text{ mm}^2$ . Height difference between the planes assumed to be $6.35 \text{ mm}$ (Figure 7.1).
Simulation Runs	6 simulation runs ( $S_1W$ , $S_1S$ , $S_2W$ , $S_2S$ , $S_3W$ and $S_3S$ ), each comprised of 1000 trials. Simulation runs $S_1W$ , $S_2W$ and $S_3W$ were conducted with the lens $L_W$ for $40 \times 40 \times 3$ , $20 \times 20 \times 3$ , and $10 \times 10 \times 3$ calibration points respectively. On the other hand, simulation runs $S_1S$ , $S_2S$ and $S_3S$ were conducted for the lens $L_S$ with equal number of calibration points.
Random Error	Uniformly distributed random error within the bound $\pm 0.1 \text{ mm}$ in each coordinated component $X$ , $Y$ and $Z$ (Figure 7.2).

Table 7.4: Simulation Results (Run  $S_1W$ )

		Heikkilä				Proposed		
		$x$	$\bar{x}_H$	$APE_H$	$SEM_H$	$\bar{x}_P$	$APE_P$	$SEM_P$
$s$		1.0000	1.0000	0.0002	0.0000	1.0000	0.0006	0.0000
$f$	$mm$	8.5000	8.4998	0.0026	0.0002	8.4999	0.0016	0.0002
$u_0$	$pixels$	650.0000	650.0012	0.0002	0.0216	650.0475	0.0073	0.0222
$v_0$	$pixels$	500.0000	500.0153	0.0031	0.0172	500.0671	0.0134	0.0179
$k_1$	$\times 10^{-3} mm^{-2}$	3.3000	3.3040	0.1208	0.0000	3.2840	0.4855	0.0000
$k_2$	$\times 10^{-5} mm^{-4}$	-2.6000	-1.5396	40.7856	0.0000	-2.5030	3.7323	0.0000
$p_1$	$\times 10^{-6} mm^{-1}$	-1.3000	-1.5731	21.0108	0.0000	-1.7750	36.5410	0.0000
$p_2$	$\times 10^{-5} mm^{-1}$	4.0000	4.2049	5.1236	0.0000	3.9344	1.6398	0.0000
$d$		0.9999	0.9999	0.0000	0.0000	0.9999	0.0000	0.0000
$a$		-0.0131	-0.0131	0.0174	0.0000	-0.0131	0.1207	0.0000
$b$		0.0043	0.0043	0.0264	0.0000	0.0043	0.3324	0.0000
$c$		0.0066	0.0066	0.0004	0.0000	0.0066	0.0035	0.0000
$t_x$	$mm$	-75.0000	-75.0000	0.0000	0.0026	-75.0058	0.0078	0.0027
$t_y$	$mm$	-55.0000	-55.0018	0.0033	0.0021	-55.0082	0.0150	0.0022
$t_z$	$mm$	220.0000	219.9923	0.0035	0.0047	220.0034	0.0016	0.0046

Table 7.5: Simulation Results (Run  $S_1S$ )

		Heikkilä				Proposed		
		$x$	$\bar{x}_H$	$APE_H$	$SEM_H$	$\bar{x}_P$	$APE_P$	$SEM_P$
$s$		1.0000	0.9993	0.0657	0.0000	1.0000	0.0005	0.0000
$f$	$mm$	8.5000	8.8316	3.9009	0.0002	8.4993	0.0078	0.0002
$u_0$	$pixels$	650.0000	662.3806	1.9047	0.0056	650.0439	0.0068	0.0186
$v_0$	$pixels$	500.0000	529.0059	5.8012	0.0075	499.9954	0.0009	0.0175
$k_1$	$\times 10^{-2} mm^{-2}$	3.3000	6.8306	106.9864	0.0000	3.2959	0.1242	0.0000
$k_2$	$\times 10^{-4} mm^{-4}$	-2.6000	-22.7041	773.2362	0.0000	-2.5660	1.3078	0.0000
$p_1$	$\times 10^{-5} mm^{-1}$	-1.3000	572.8647	44166.5125	0.0000	-1.3426	3.2802	0.0000
$p_2$	$\times 10^{-4} mm^{-1}$	4.0000	31.3140	682.8503	0.0000	4.0464	1.1605	0.0000
$d$		0.9999	1.0000	0.0089	0.0000	0.9999	0.0000	0.0000
$a$		-0.0131	-0.0036	72.6557	0.0000	-0.0131	0.0119	0.0000
$a$		0.0043	-0.0006	114.8913	0.0000	0.0043	0.2344	0.0000
$c$		0.0066	0.0065	1.2659	0.0000	0.0066	0.0031	0.0000
$t_x$	$mm$	-75.0000	-76.4991	1.9987	0.0007	-75.0053	0.0071	0.0023
$t_y$	$mm$	-55.0000	-58.4735	6.3155	0.0009	-54.9996	0.0008	0.0021
$t_z$	$mm$	220.0000	219.0583	0.4281	0.0046	220.0014	0.0006	0.0045

Table 7.6: Simulation Results (Run  $S_2W$ )

			Heikkila			Proposed		
		$x$	$\bar{x}_H$	$APE_H$	$SEM_H$	$\bar{x}_P$	$APE_P$	$SEM_P$
$s$		1.0000	1.0000	0.0002	0.0000	1.0000	0.0006	0.0000
$f$	$mm$	8.5000	8.4998	0.0026	0.0002	8.4999	0.0016	0.0002
$u_0$	$pixels$	650.0000	650.0012	0.0002	0.0216	650.0475	0.0073	0.0222
$v_0$	$pixels$	500.0000	500.0153	0.0031	0.0172	500.0671	0.0134	0.0179
$k_1$	$\times 10^{-3} mm^{-2}$	3.3000	3.3055	0.1655	0.0000	3.2820	0.5457	0.0000
$k_2$	$\times 10^{-5} mm^{-4}$	-2.6000	-1.5458	40.5452	0.0000	-2.4876	4.3249	0.0000
$p_1$	$\times 10^{-6} mm^{-1}$	-1.3000	-1.9147	47.2815	0.0000	-2.3115	77.8059	0.0000
$p_2$	$\times 10^{-5} mm^{-1}$	4.0000	4.3088	7.7212	0.0000	3.9396	1.5112	0.0000
$d$		0.9999	0.9999	0.0000	0.0000	0.9999	0.0000	0.0000
$a$		-0.0131	-0.0131	0.0174	0.0000	-0.0131	0.1207	0.0000
$a$		0.0043	0.0043	0.0264	0.0000	0.0043	0.3324	0.0000
$c$		0.0066	0.0066	0.0004	0.0000	0.0066	0.0035	0.0000
$t_x$	$mm$	-75.0000	-75.0000	0.0000	0.0026	-75.0058	0.0078	0.0027
$t_y$	$mm$	-55.0000	-55.0018	0.0033	0.0021	-55.0082	0.0150	0.0022
$t_z$	$mm$	220.0000	219.9923	0.0035	0.0047	220.0034	0.0016	0.0046



Table 7.7: Simulation Results (Run  $S_2S$ )

		Heikkilä				Proposed		
		$x$	$\bar{x}_H$	$APE_H$	$SEM_H$	$\bar{x}_P$	$APE_P$	$SEM_P$
$s$		1.0000	0.9993	0.0732	0.0000	1.0000	0.0004	0.0000
$f$	$mm$	8.5000	8.8247	3.8197	0.0004	8.4991	0.0102	0.0003
$u_0$	$pixels$	650.0000	652.7904	0.4293	0.0109	650.0735	0.0113	0.0389
$v_0$	$pixels$	500.0000	523.7585	4.7517	0.0148	499.9767	0.0047	0.0356
$k_1$	$\times 10^{-2} mm^{-2}$	3.3000	6.8006	106.0781	0.0000	3.2957	0.1317	0.0000
$k_2$	$\times 10^{-4} mm^{-4}$	-2.6000	-23.0898	788.0683	0.0000	-2.5635	1.4040	0.0000
$p_1$	$\times 10^{-5} mm^{-1}$	-1.3000	465.6876	35922.1222	0.0000	-1.6155	24.2698	0.0000
$p_2$	$\times 10^{-4} mm^{-1}$	4.0000	12.2026	205.0649	0.0000	4.0812	2.0303	0.0000
$d$		0.9999	1.0000	0.0077	0.0000	0.9999	0.0000	0.0000
$a$		-0.0131	-0.0054	58.9805	0.0000	-0.0131	0.0109	0.0000
$a$		0.0043	0.0028	34.7248	0.0000	0.0043	0.3821	0.0000
$c$		0.0066	0.0066	0.1336	0.0000	0.0066	0.0148	0.0000
$t_z$	$mm$	-75.0000	-75.3390	0.4520	0.0013	-75.0091	0.0121	0.0048
$t_y$	$mm$	-55.0000	-57.8531	5.1874	0.0018	-54.9972	0.0052	0.0043
$t_x$	$mm$	220.0000	219.5744	0.1934	0.0091	219.9968	0.0014	0.0087

Table 7.8: Simulation Results (Run  $S_3W$ )

			Heikkilä			Proposed		
		$x$	$\bar{x}_H$	$APE_H$	$SEM_H$	$\bar{x}_P$	$APE_P$	$SEM_P$
$s$		1.0000	1.0000	0.0006	0.0000	1.0000	0.0001	0.0000
$f$	$mm$	8.5000	8.5002	0.0029	0.0007	8.5012	0.0144	0.0007
$u_0$	$pixels$	650.0000	650.1455	0.0224	0.0906	649.9109	0.0137	0.0909
$v_0$	$pixels$	500.0000	500.0756	0.0151	0.0715	500.0038	0.0008	0.0699
$k_1$	$\times 10^{-3} mm^{-2}$	3.3000	3.3051	0.1547	0.0000	3.2852	0.4495	0.0000
$k_2$	$\times 10^{-5} mm^{-4}$	-2.6000	-1.5473	40.4896	0.0000	-2.4990	3.8849	0.0000
$p_1$	$\times 10^{-6} mm^{-1}$	-1.3000	-1.4656	12.7347	0.0000	-0.6087	53.1755	0.0000
$p_2$	$\times 10^{-5} mm^{-1}$	4.0000	4.1723	4.3086	0.0000	3.9678	0.8039	0.0000
$d$		0.9999	0.9999	0.0000	0.0000	0.9999	0.0001	0.0000
$a$		-0.0131	-0.0131	0.0997	0.0000	-0.0131	0.0955	0.0000
$a$		0.0043	0.0043	0.5810	0.0000	0.0043	0.3690	0.0000
$c$		0.0066	0.0066	0.0081	0.0000	0.0066	0.0019	0.0000
$t_x$	$mm$	-75.0000	-75.0171	0.0227	0.0110	-74.9892	0.0144	0.0110
$t_y$	$mm$	-55.0000	-55.0092	0.0167	0.0086	-55.0002	0.0004	0.0085
$t_z$	$mm$	220.0000	219.9985	0.0007	0.0181	220.0429	0.0195	0.0184

Table 7.9: Simulation Results (Run  $S_3S$ )

		Heikkila				Proposed		
		$x$	$\bar{x}_H$	$APE_H$	$SEM_H$	$\bar{x}_P$	$APE_P$	$SEM_P$
$s$		1.0000	0.9993	0.0732	0.0000	1.0000	0.0004	0.0000
$f$	$mm$	8.5000	8.8247	3.8197	0.0004	8.4991	0.0102	0.0003
$u_0$	$pixels$	650.0000	652.7904	0.4293	0.0109	650.0735	0.0113	0.0389
$v_0$	$pixels$	500.0000	523.7585	4.7517	0.0148	499.9767	0.0047	0.0356
$k_1$	$\times 10^{-2} mm^{-2}$	3.3000	6.6814	102.4658	0.0000	3.2968	0.0977	0.0000
$k_2$	$\times 10^{-4} mm^{-4}$	-2.6000	-24.3565	836.7880	0.0000	-2.5710	1.1149	0.0000
$p_1$	$\times 10^{-5} mm^{-1}$	-1.3000	114.6791	8921.4729	0.0000	-1.2116	6.7996	0.0000
$p_2$	$\times 10^{-4} mm^{-1}$	4.0000	-37.2873	1032.1813	0.0000	4.0736	1.8388	0.0000
$d$		0.9999	1.0000	0.0077	0.0000	0.9999	0.0000	0.0000
$a$		-0.0131	-0.0054	58.9805	0.0000	-0.0131	0.0109	0.0000
$a$		0.0043	0.0028	34.7248	0.0000	0.0043	0.3821	0.0000
$c$		0.0066	0.0066	0.1336	0.0000	0.0066	0.0148	0.0000
$t_x$	$mm$	-75.0000	-75.3390	0.4520	0.0013	-75.0091	0.0121	0.0048
$t_y$	$mm$	-55.0000	-57.8531	5.1874	0.0018	-54.9972	0.0052	0.0043
$t_z$	$mm$	220.0000	219.5744	0.1934	0.0091	219.9968	0.0014	0.0087

Table 7.10: Comparison of the Estimates of the Intrinsic Camera Parameters Based on  $APE$ 

	$S_1W$		$S_2W$		$S_3W$		$S_1S$		$S_2S$		$S_3S$	
	$APE_H$	$APE_P$	$APE_H$	$APE_P$	$APE_H$	$APE_P$	$APE_H$	$APE_P$	$APE_H$	$APE_P$	$APE_H$	$APE_P$
$s$	0.00	0.00	0.00	0.00	0.00	0.00	0.07	0.00	0.07	0.00	0.17	0.00
$f$	0.00	0.00	0.00	0.01	0.00	0.01	3.90	0.01	3.82	0.01	3.27	0.01
$u_0$	0.00	0.01	0.00	0.01	0.02	0.01	1.90	0.01	0.43	0.01	3.47	0.01
$v_0$	0.00	0.01	0.01	0.02	0.02	0.00	5.80	0.00	4.75	0.00	1.28	0.00
$k_1$	0.12	0.49	0.17	0.55	0.15	0.45	106.99	0.12	106.08	0.13	102.47	0.10
$k_2$	40.79	3.73	40.55	4.32	40.49	3.88	773.24	1.31	788.07	1.40	836.79	1.11
$p_1$	21.01	36.54	47.28	77.81	12.73	53.18	44166.51	3.28	35922.12	24.27	8921.47	6.80
$p_2$	5.12	1.64	7.72	1.51	4.31	0.80	682.85	1.16	205.06	2.03	1032.18	1.84



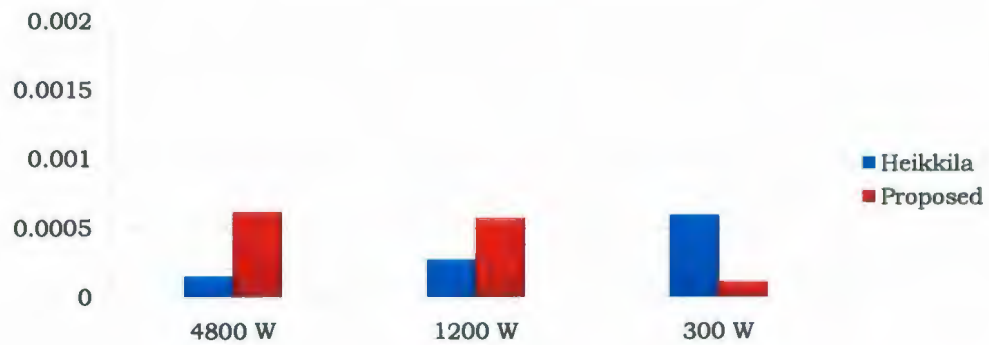


Figure 7.4: *APE* of the Estimates of the Scale Factor  $s$  ( $L_W$ )

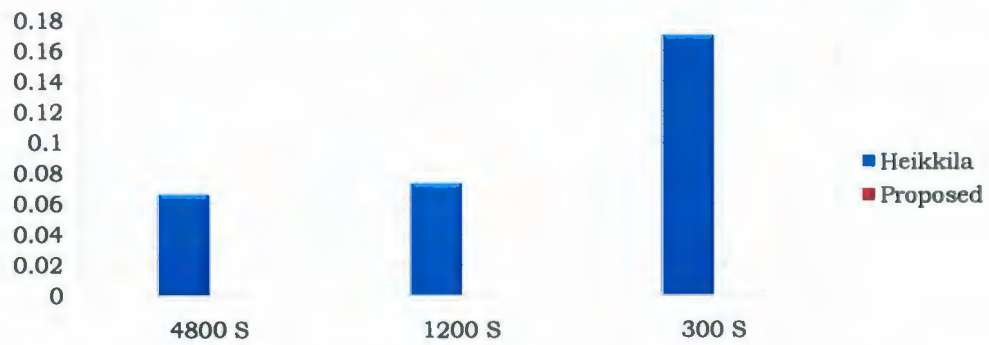


Figure 7.5: *APE* of the Estimates of the Scale Factor  $s$  ( $L_S$ )

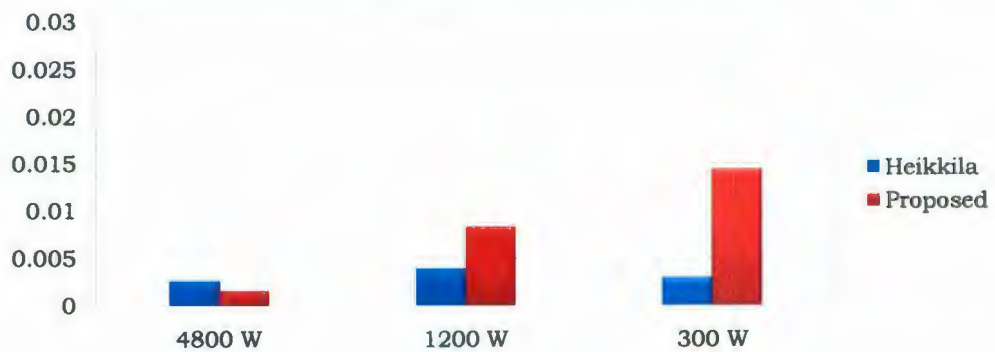


Figure 7.6: *APE* of the Estimates of the Effective Focal Length  $f$  ( $L_W$ )

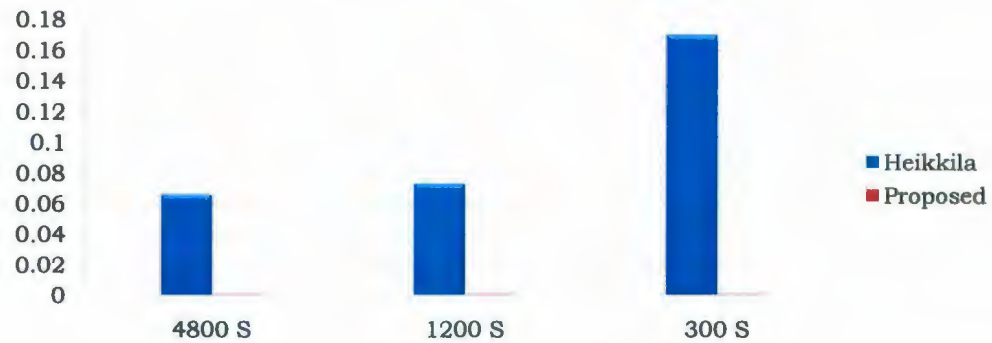


Figure 7.7: *APE* of the Estimates of the Effective Focal Length  $f$  ( $L_S$ )

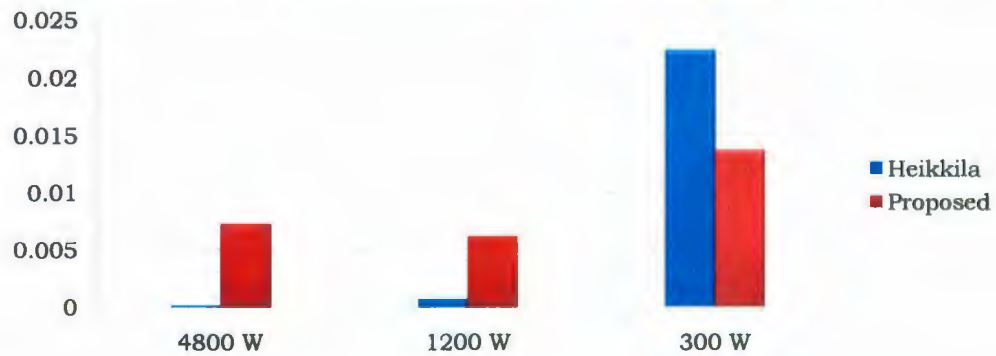


Figure 7.8: *APE* of the Estimates of the Image Center Coordinate  $u_0$  ( $L_W$ )

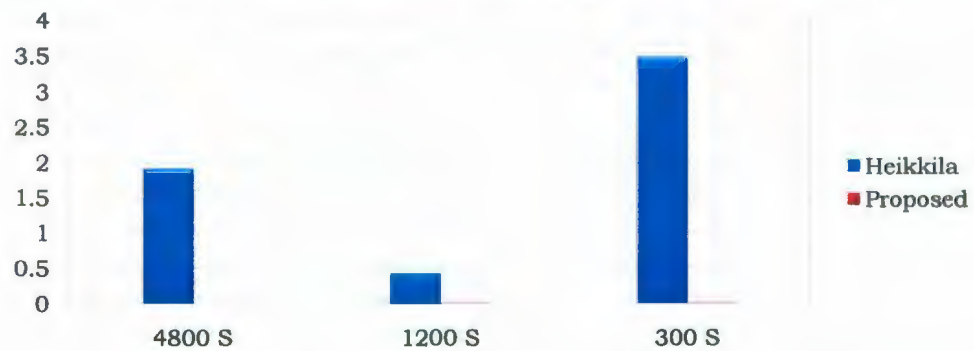


Figure 7.9: *APE* of the Estimates of the Image Center Coordinate  $u_0$  ( $L_S$ )

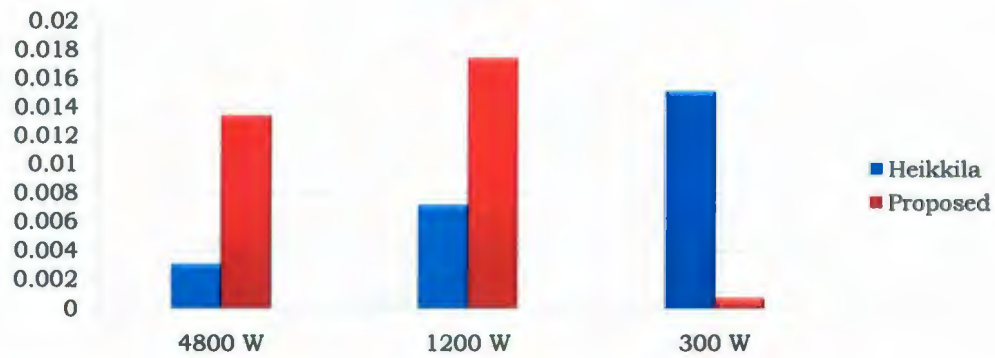


Figure 7.10: *APE* of the Estimates of the Image Center Coordinate  $v_0$  ( $L_W$ )

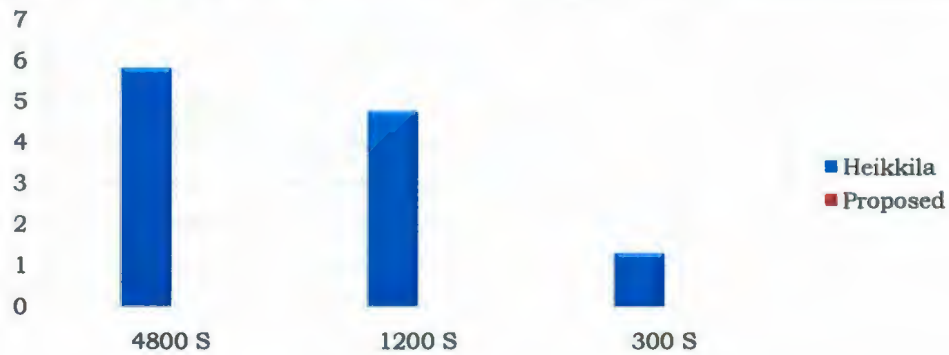


Figure 7.11: *APE* of the Estimates of the Image Center Coordinate  $v_0$  ( $L_S$ )

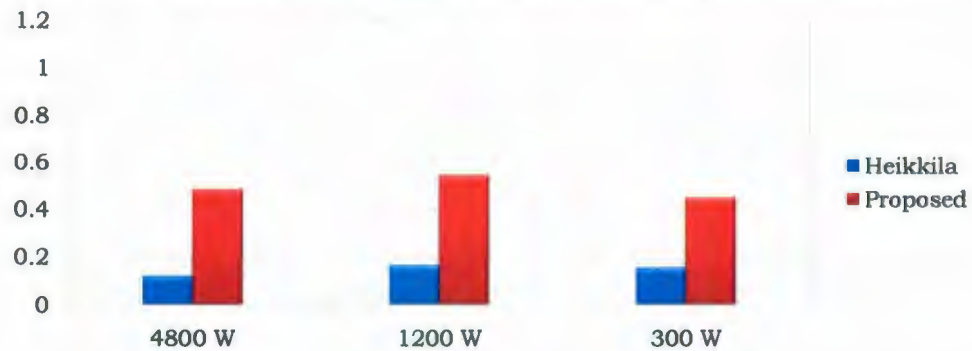


Figure 7.12: *APE* of the Estimates of the Radial Lens Distortion Coefficient  $k_1$  ( $L_W$ )



Figure 7.13: *APE* of the Estimates of the Radial Lens Distortion Coefficient  $k_1$  ( $L_S$ )

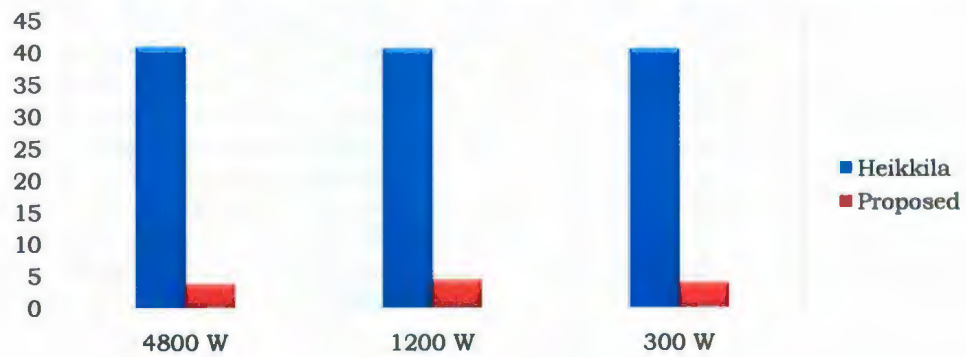


Figure 7.14: *APE* of the Estimates of the Radial Lens Distortion Coefficient  $k_2$  ( $L_W$ )

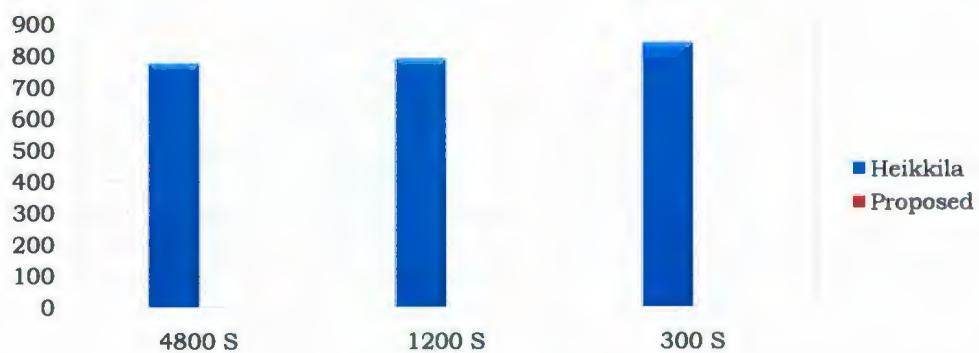


Figure 7.15: *APE* of the Estimates of the Radial Lens Distortion Coefficient  $k_2$  ( $L_S$ )



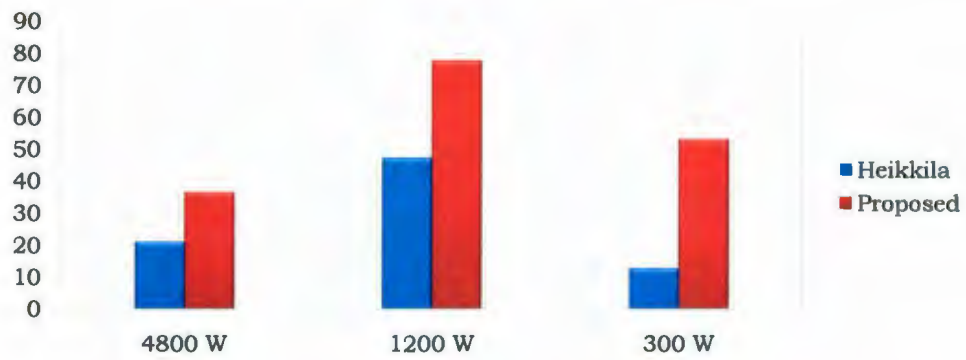


Figure 7.16: *APE* of the Estimates of the Radial Lens Distortion Coefficient  $p_1$  ( $L_W$ )



Figure 7.17: *APE* of the Estimates of the Radial Lens Distortion Coefficient  $p_1$  ( $L_S$ )

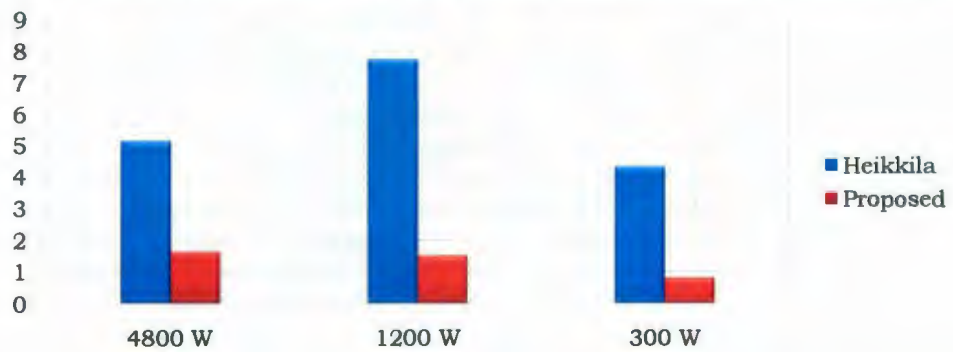


Figure 7.18: *APE* of the Estimates of the Radial Lens Distortion Coefficient  $p_2$  ( $L_W$ )

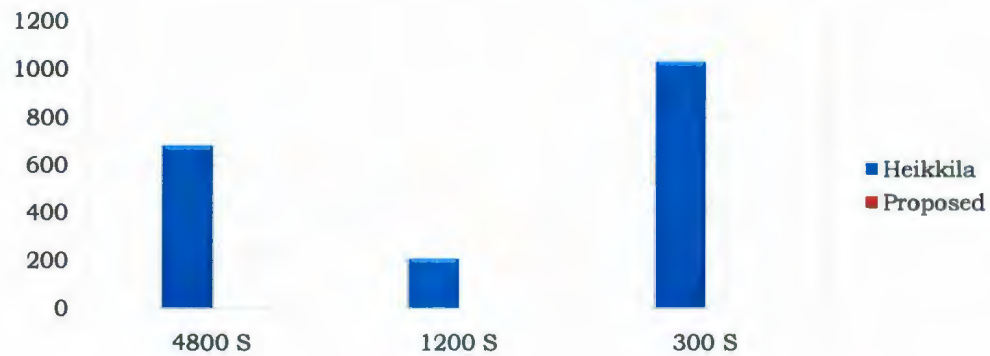


Figure 7.19: *APE* of the Estimates of the Radial Lens Distortion Coefficient  $p_2$  ( $L_S$ )



Figure 7.20: *APE* of the Estimates of the Quaternion Component  $d$  ( $L_W$ )



Figure 7.21: *APE* of the Estimates of the Quaternion Component  $d$  ( $L_S$ )

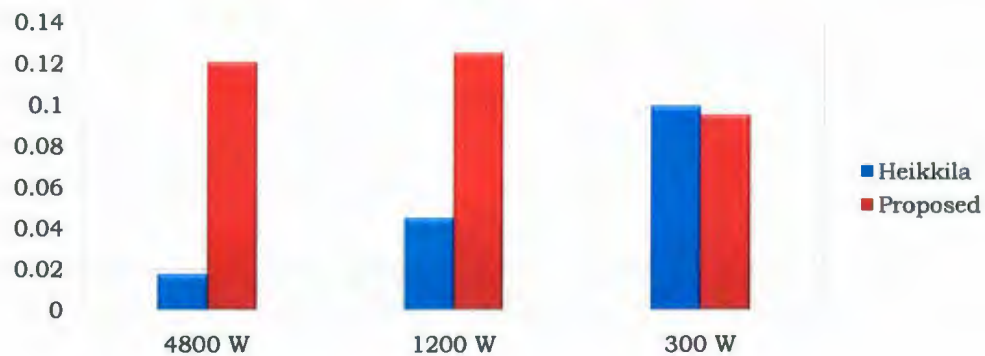


Figure 7.22: *APE* of the Estimates of the Quaternion Component  $a$  ( $L_W$ )

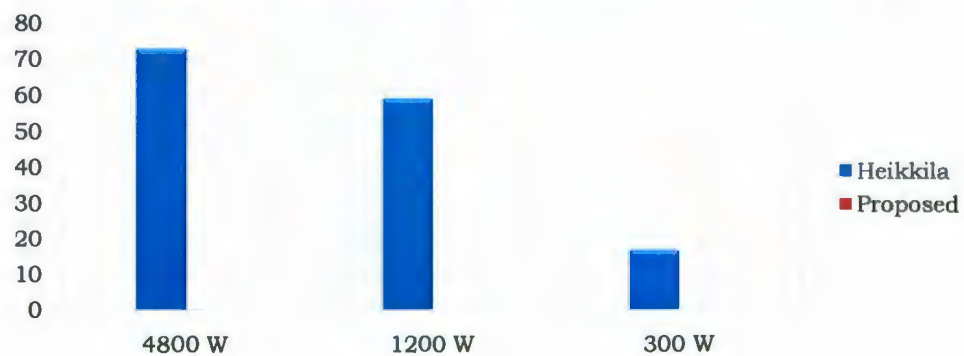


Figure 7.23: *APE* of the Estimates of the Quaternion Component  $a$  ( $L_S$ )

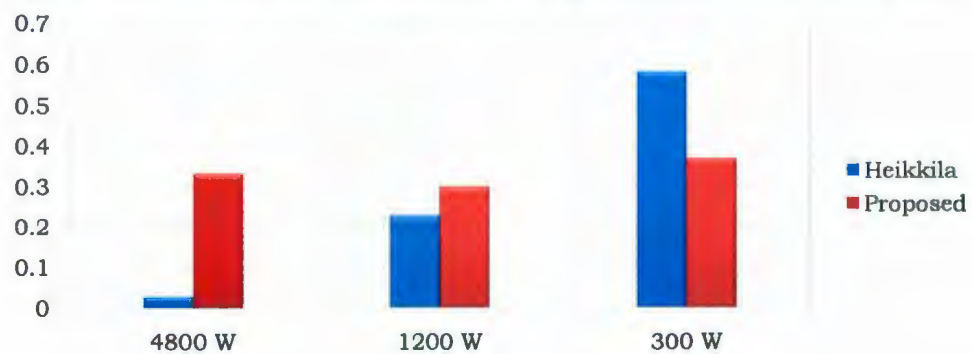


Figure 7.24: *APE* of the Estimates of the Quaternion Component  $b$  ( $L_W$ )

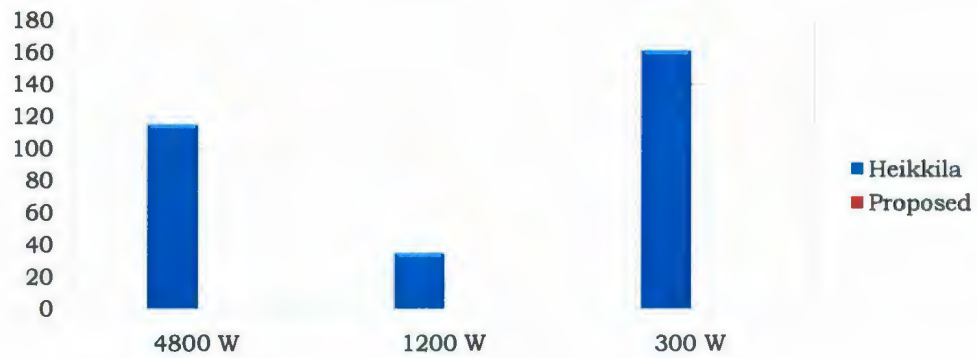


Figure 7.25: *APE* of the Estimates of the Quaternion Component  $b$  ( $L_S$ )

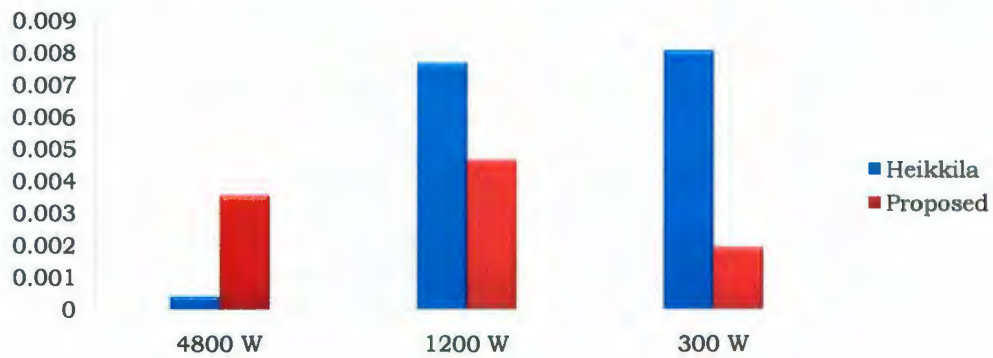


Figure 7.26: *APE* of the Estimates of the Quaternion Component  $c$  ( $L_W$ )

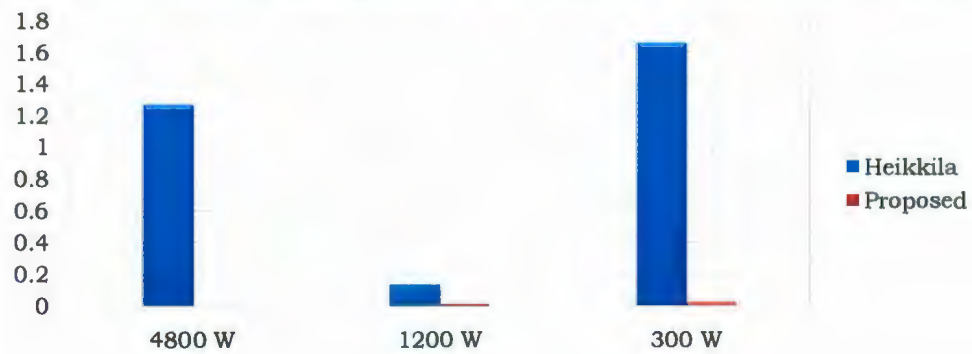


Figure 7.27: *APE* of the Estimates of the Quaternion Component  $c$  ( $L_S$ )



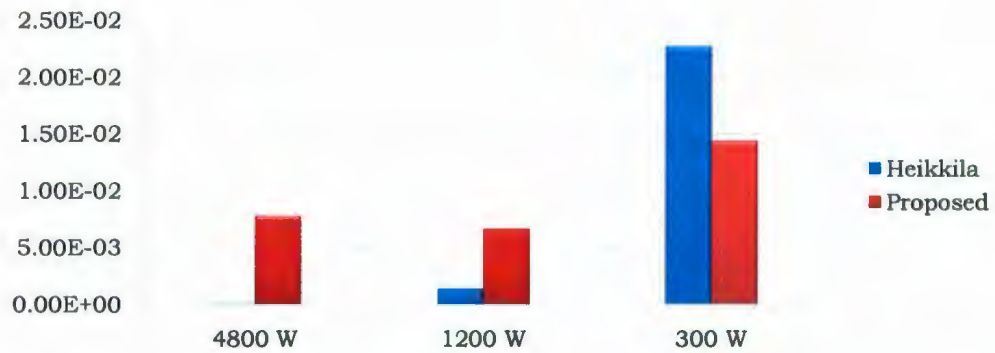


Figure 7.28: *APE* of the Estimates of the Extrinsic Position Vector Component  $t_x$

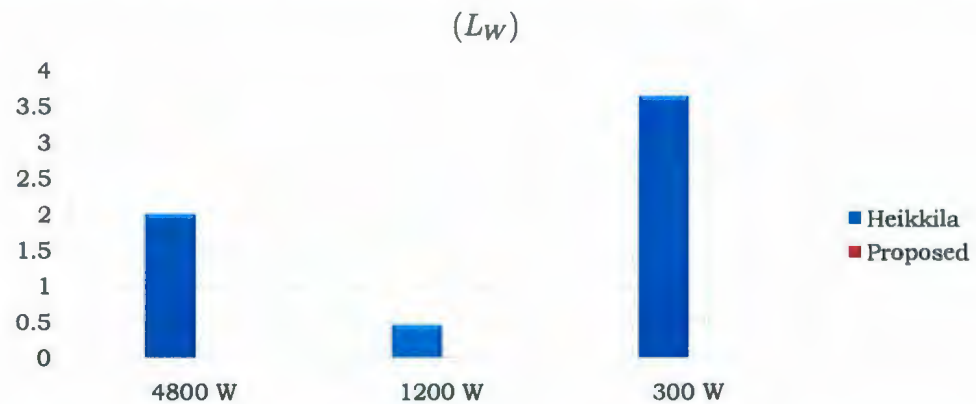


Figure 7.29: *APE* of the Estimates of the Extrinsic Position Vector Component  $t_x$

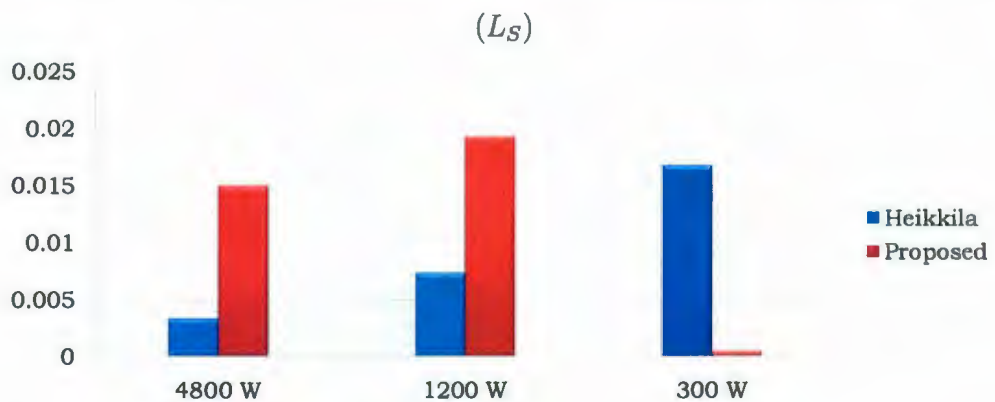


Figure 7.30: *APE* of the Estimates of the Extrinsic Position Vector Component  $t_y$

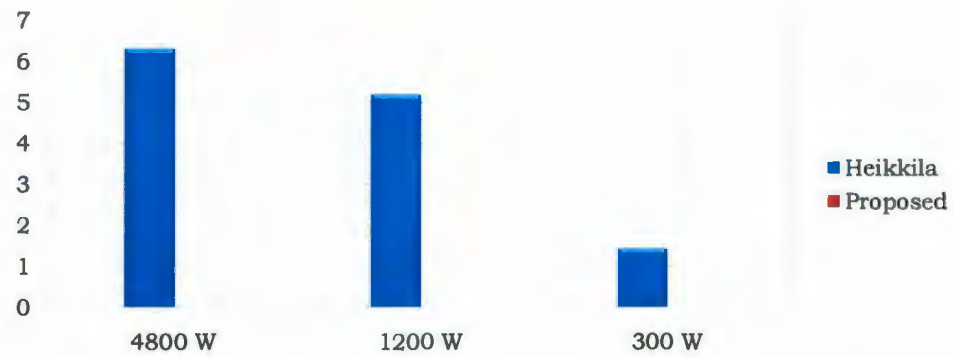


Figure 7.31: *APE* of the Estimates of the Extrinsic Position Vector Component  $t_y$  ( $L_S$ )

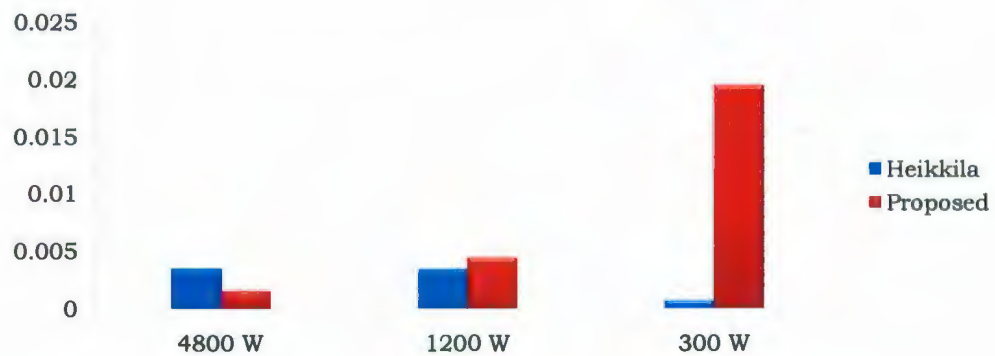


Figure 7.32: *APE* of the Estimates of the Extrinsic Position Vector Component  $t_z$  ( $L_W$ )

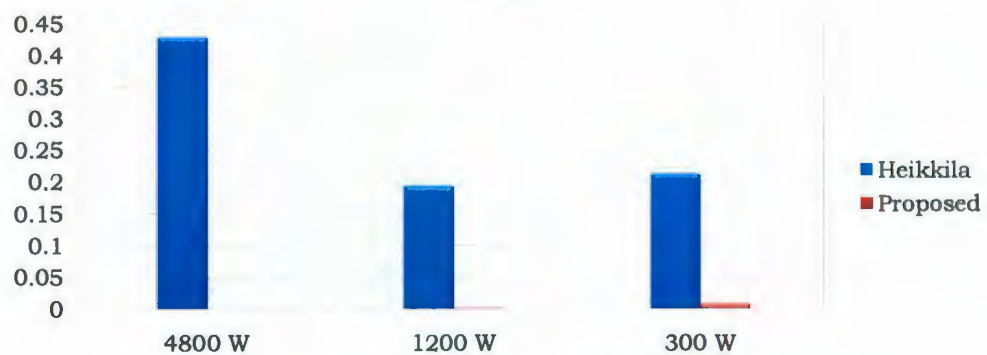


Figure 7.33: *APE* of the Estimates of the Extrinsic Position Vector Component  $t_z$  ( $L_S$ )

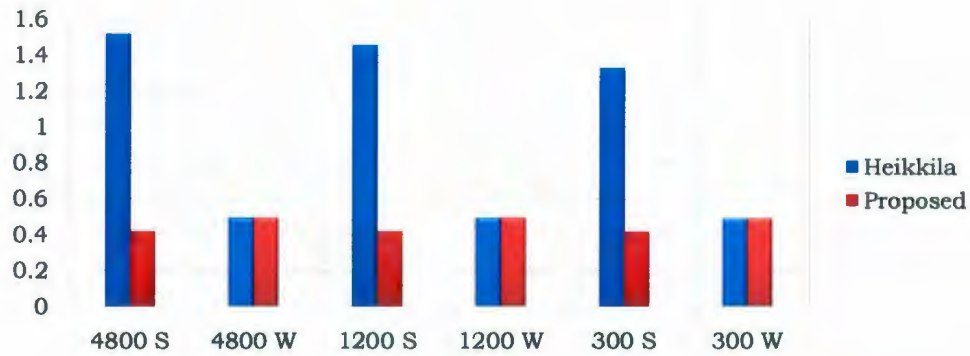


Figure 7.34: Average Standard Error in Pixels ( $X$  Axis)

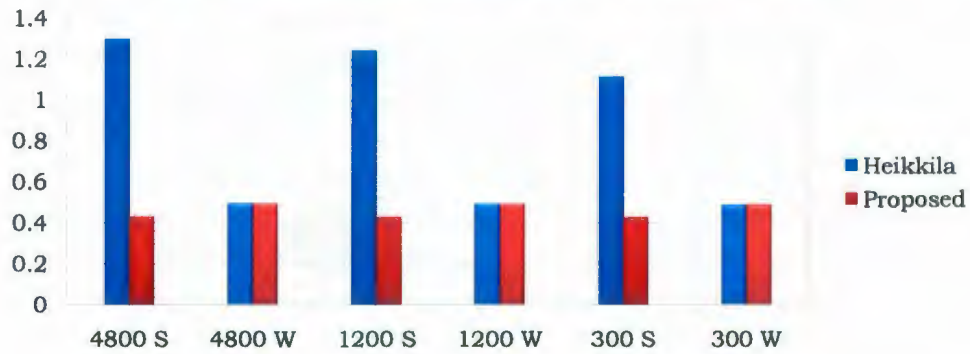


Figure 7.35: Average Standard Error in Pixels ( $Y$  Axis)

#### 7.1.4 Inferences from the Simulation Results

In Tables 7.4, 7.5, 7.6, 7.7, 7.8 and 7.9 the simulation results provided by runs  $S_1W$ ,  $S_1S$ ,  $S_2W$ ,  $S_2S$ ,  $S_3W$  and  $S_3S$ ) are presented in the stated order. In Figures 7.4-7.33 the  $APEs$  of the different camera parameters provided by Heikkila's technique and the proposed technique are presented graphically for comprehensive interpretation of the data.

A review of the data presented in the tables shows that in contrast to the  $APEs$

Table 7.11: Comparison of the Standard Errors in Pixels (Simulated Camera Calibration)

	<i>X</i> Axis		<i>Y</i> Axis	
	<i>H</i>	<i>P</i>	<i>H</i>	<i>P</i>
$S_1W$	0.4940	0.4941	0.4933	0.4934
$S_2W$	0.4927	0.4932	0.4922	0.4921
$S_3W$	0.4888	0.4894	0.4879	0.4884
$S_1S$	1.5199	0.4184	1.2988	0.4303
$S_2S$	1.4579	0.4178	1.2436	0.4298
$S_3S$	1.3307	0.4151	1.1161	0.4271

of the extrinsic camera parameters provided by Heikkila's technique, the proposed technique performs better especially in presence of strong lens distortion effect. When the lens distortion effect is weak, both of the techniques produce comparable accuracy.

The *APEs* of the estimates of the intrinsic parameters provided by both techniques are compiled together in Table 7.10. Studying the data presented in Table 7.10 reveals the following:

- For the lens  $L_W$  (i.e., in the presence of weak lens distortion effect), the estimates of the parameters  $s$ ,  $f$ ,  $u_0$ ,  $v_0$  and  $k_1$  provided by both techniques are similarly accurate.
- The accuracies of the estimates of the parameters  $k_2$  and  $p_2$  with weak lens distortion effect provided by the proposed technique are comprehensively better than those provided by Heikkila's technique.



- The estimates of the tangential lens distortion coefficient  $p_1$  provided by the proposed technique for the lens  $L_W$  are worse than that provided by Heikkila's technique. This can be attributed to the large random error which had a greater impact on the imaging process than the parameter itself.
- For the estimates of all the lens distortion coefficient  $(k_1, k_2, p_1, p_2)$  and the parameters  $u_0$ ,  $v_0$  and  $f$  under strong lens distortion effect ( $L_S$ ), the proposed techniques exhibit an inclusively higher order accuracy than Heikkila's technique. This can be attributed to the Taylor series approximation of the lens distortion model in Heikkila's camera model [4]. The proposed technique does not approximate the lens distortion model; rather the minimization of the error is conducted in the undistorted space which consequently produced better accuracy in the estimates of those particular camera parameters. Greater interaction between the image center, effective focal length and the lens distortion coefficients resulted in larger errors in the estimates of the image center and the effective focal length yielded by Heikkila's technique.

Upon compiling the aforementioned observations one can conclude with fair justification that the proposed technique is more robust in terms of accurately estimating the camera intrinsic and extrinsic parameters regardless of the amount of lens distortion present in the imaging system.

The standard errors in pixels provided by both of the calibration techniques are graphically presented in Figures 7.34 and 7.35 and listed in Table 7.11. In agreement with the previous observations, Heikkila's calibration technique produces more than three times as much standard error in pixels provided by the proposed technique

when strong lens distortion effect is present. From the standpoint of a practical calibration scenario, this is particularly attractive since the proposed technique can better minimize the discrepancy between the practical observations and reconstructed observations regardless of the optical properties of the lens.

## 7.2 Accuracy in Practical Camera Calibration

Besides computer simulation, an industrial grade digital camera was calibrated using the proposed calibration technique to assess the performance of the camera calibration. Additionally Heikkila's technique was also used to calibrate the camera based on the identical calibration data for benchmarking purposes. The calibration data were acquired by the method described in Chapter 6. In Table 7.12 the calibrated camera models are presented. It can be easily seen that the camera models estimated by the proposed technique and Heikkila's technique is almost similar. This observation is consistent with the computer simulation for a camera with a lens with weak distortion effect.

The residuals of the calibrated camera models are presented in Figure 7.36.

### 7.2.1 Accuracy Evaluation

Armangué *et al.* in [10] reviewed major calibration techniques with accuracy assessment based on practical camera calibration. The evaluation criteria were both 3D and 2D measurements of known world points and the corresponding measured image points. In this work only 2D measurements were adopted as a means of accuracy evaluation. These measurements were carried out in the distorted space and as well

Table 7.12: Comparison of the Standard Errors in Pixels

Parameter		Proposed		Heikkila	
		Unrefined	Refined	Unrefined	Refined
$s$		1.0018	1.0017	1.0018	1.0017
$f$	$mm$	36.6594	36.6708	36.6597	36.6738
$k_1$	$\times 10^{-4} mm^{-2}$	-1.4368	-1.6453	-1.3854	-1.5913
$k_2$	$\times 10^{-6} mm^{-4}$	1.1082	1.3280	0.7445	0.9592
$p_1$	$\times 10^{-4} mm^{-1}$	-1.8052	-1.9782	-1.8371	-2.0076
$p_2$	$\times 10^{-6} mm^{-1}$	9.2414	-25.4319	8.5677	-25.3948
$u_0$	$pixels$	755.7412	772.2868	756.2747	772.4660
$v_0$	$pixels$	628.4292	635.7743	630.3798	637.5887
$d$		0.9999	0.9999	0.9999	0.9999
$a$		0.0132	0.0137	0.0133	0.0138
$a$		-0.0027	-0.0038	-0.0027	-0.0038
$c$		-0.0048	-0.0048	-0.0048	-0.0048
$t_x$	$mm$	-65.6299	-67.5439	-65.6915	-67.5645
$t_y$	$mm$	-54.0298	-54.8800	-54.2559	-55.0902
$t_z$	$mm$	913.0139	913.2917	912.9815	913.3282
Standard Error in Pixels ( $X$ & $Y$ )					
$\epsilon_x$	$pixels$	0.3041	0.3107	0.3041	0.3107
$\epsilon_y$	$pixels$	0.1761	0.1914	0.1761	0.1913



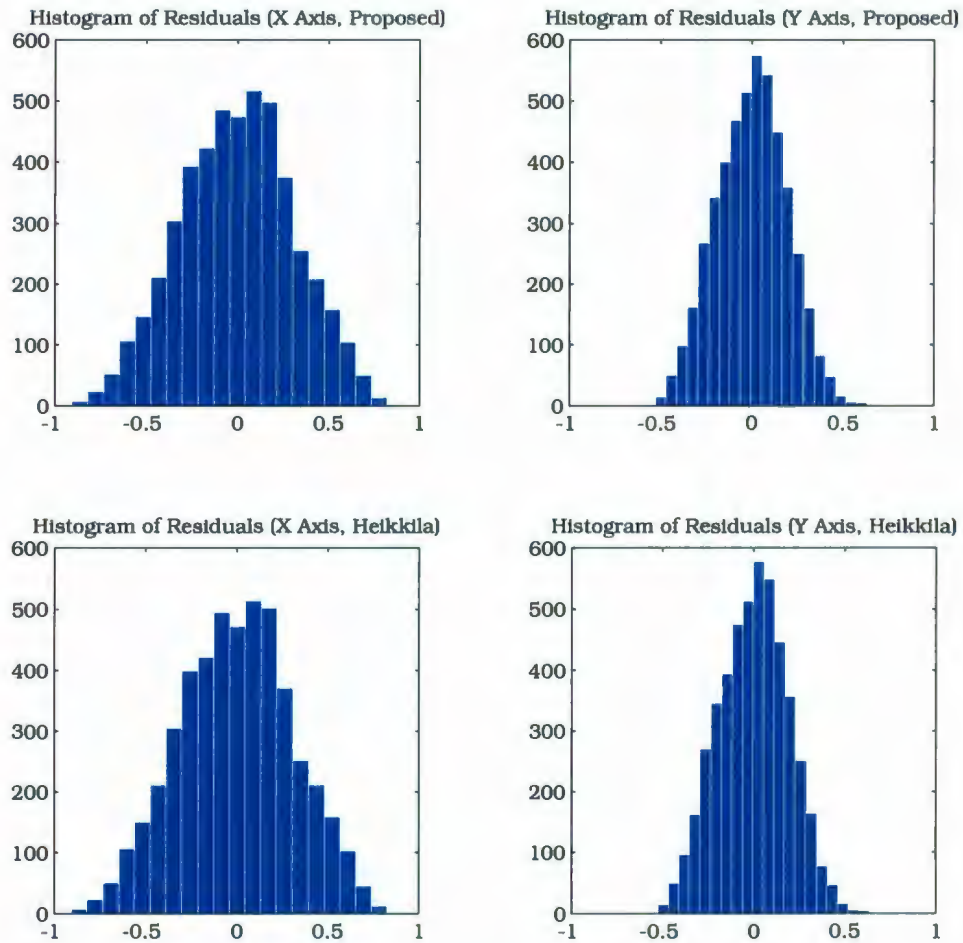


Figure 7.36: Residuals of the Calibrated Camera Models (Proposed & Heikkila)

as the undistorted space:

- *Distorted Measurement:* The distorted image points of the known world points are calculated according to the calibrated camera model. The discrepancy between the estimated distorted image points and the measured image points



provides an index for accuracy of the calibration technique.

- *Undistorted Measurement*: The undistorted image points are calculated by removing distortion from the corresponding measured image points using the calibrated lens distortion model. The ray traced undistorted image points of the known world points are subsequently calculated according to the calibrated camera model under pure perspective projection. The distances between the undistorted versions of the measured image points and the reconstructed undistorted image points quantify the accuracy of the calibration.

In Table 7.13 the 2D measurements of the calibrated camera models (Table 7.12) are presented. Since the calibrated camera models by both techniques are almost similar, the measurement data are also consistent with the previous observation.

Table 7.13: 2D Measurements (*pixels*)

2D Undistorted Measurements		2D Distorted Measurements	
	Proposed	Heikkila	
	Proposed	Heikkila	
$\mu$	0.3195	0.3194	0.4602
$\sigma$	0.1759	0.1759	0.2839
$max$	0.9906	0.9898	1.7058

## Chapter 8

# Surface Reconstruction

Retrieving 3D metric information from image data is the main focus of this work. When 3D world is projected on to an image plane, the so called depth or range information is no longer available from the image input only. For example, in Figure 8.1 the points  $P$  and  $Q$  in 3-Space are imaged as the same point  $P'$  on the image plane. Even with a calibrated camera it is not possible to determine the ranges of the points  $P$  and  $Q$  from their common image point  $P'$ . This phenomenon is often referred to as loss of information of depth during perspective projection. The loss of depth or range information can be generally termed as the *reconstruction* problem that deals with acquiring a 3D model of the scene from the image. From Figure 8.1 it is clear that a single image of a point in 3-Space is not sufficient for unique localization of that point in 3-Space by back projection. Hence additional sensing techniques are required to complement the image data. One such sensing technique is the stereoscopic approach where two or more cameras are employed to acquire multiple images of the object to retrieve the range data by triangulation. But due to the correspondence problem

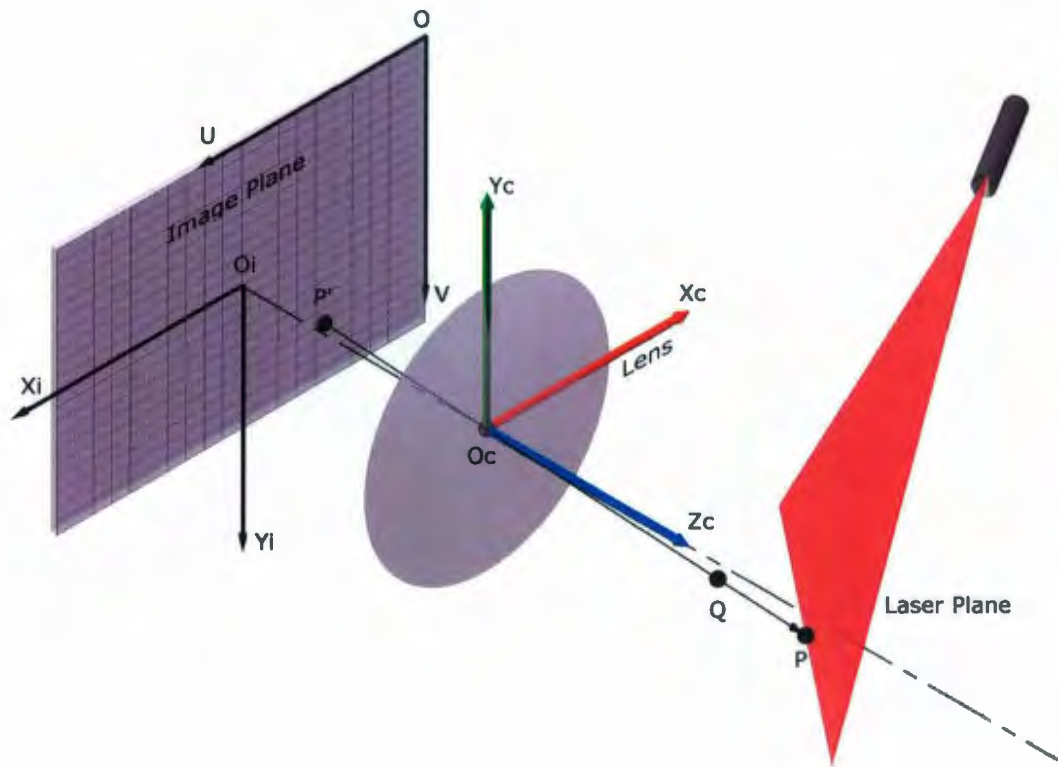


Figure 8.1: Range Ambiguity in Perspective Projection

in the images, this approach usually yields inaccurate reconstruction of the scene at a high computational cost. Besides the stereoscopic approach, other reconstruction techniques include shape from shading [48], shape from motion [49], shape from texture [50], etc. Unfortunately these techniques do not provide the desired accuracy. Please note that, these techniques are also called *passive* techniques since the involved reconstruction system does not interact with the object. Whereas *active* techniques use specialized illumination sources and detectors to overcome the fundamental ambiguities associated with the passive techniques [51]. Some active techniques use laser radar [52, 53], others use various forms of *Structured Light* [54, 55].

In this work to solve the problem of reconstruction input from a structured light source (laser projector emitting a linear pattern) is used to complement the image data to retrieve the range of the object by triangulation.

## 8.1 Basic Principle of Structured Light Based Reconstruction

The camera needs to be calibrated to provide a complete geometrical interpretation of the image formation process prior to the range determination by triangulation is attempted. Let the image point  $P'$  be defined by the coordinates  $[u_d v_d]^T$  with respect to the image coordinate frame  $UV$ . The image coordinates  $[u_d v_d]^T$  are essentially the distorted image coordinates of the image point  $P'$ . The lens distortion present in the image can be removed by refining the distorted image coordinates  $[u_d v_d]^T$  to provide the undistorted image coordinates  $[u v]^T$  according to the calibrated lens distortion model.

The undistorted image coordinates  $[u v]^T$  of the image point  $P'$  provides the direction vector of the light ray  $P'O_cP$  (Figure 8.1). Let a point be on the light ray  $P'O_cP$  whose coordinates are  $[x_c y_c f]^T$ . From Equation 2.6 it can easily shown that  $x_c = \frac{(u-u_0)}{s}$  and  $y_c = (v - v_0)$ . The image coordinates are usually expressed in pixel units. In order to express the image coordinates in absolute unit conversion factors  $\lambda_x$  and  $\lambda_y$  are used (Equation 8.1).  $\lambda_x$  and  $\lambda_y$  are the lengths of the pixel in the



horizontal and vertical direction respectively.

$$\begin{bmatrix} x_c \\ y_c \\ z_c \end{bmatrix} = \begin{bmatrix} \frac{(u-u_0)}{s} \lambda_x \\ (v-v_0) \lambda_y \\ f \end{bmatrix} \quad (8.1)$$

The position vector in Equation 8.1 can be normalized to yield the unit direction vector representing the light ray associated with the undistorted image point  $[u \ v]^T$ . Any point on this light ray can be represented by the general position vector  $\tau[x_c \ y_c \ z_c]^T$ . Here  $\tau$  is a scalar that uniquely defines the position of a particular point on the light ray. To determine the position of the point  $P$  in 3-Space uniquely the corresponding value of  $\tau$  must be determined first. If it is known that the point  $P$  lies in the laser plane, from the position and orientation of the laser plane the value of  $\tau$  corresponding to the position vector of  $P$  can be determined.

Let the equation of the laser plane with respect to the camera coordinate frame be  $n_x x + n_y y + n_z z + n_d = 0$ . Since the point  $P$  lies in this plane Equation 8.2 must also hold.

$$\tau(n_x x_c + n_y y_c + n_z z_c) + n_d = 0 \quad (8.2)$$

Solving Equation 8.2 for  $\tau$  yields:

$$\tau = \frac{-n_d}{n_x x_c + n_y y_c + n_z z_c} \quad (8.3)$$

Hence Equation 8.3 yields the value of  $\tau$  that uniquely defines the position of point  $P$ . This provides an effective solution for the reconstruction problem.

Please note that, *a priori* knowledge of the position and the orientation of the laser plane must be available before the reconstruction problem is attempted. The task of localizing the laser plane in 3-Space with respect to the camera coordinate

frame is called laser calibration. In the following section the methodology for laser calibration is discussed.

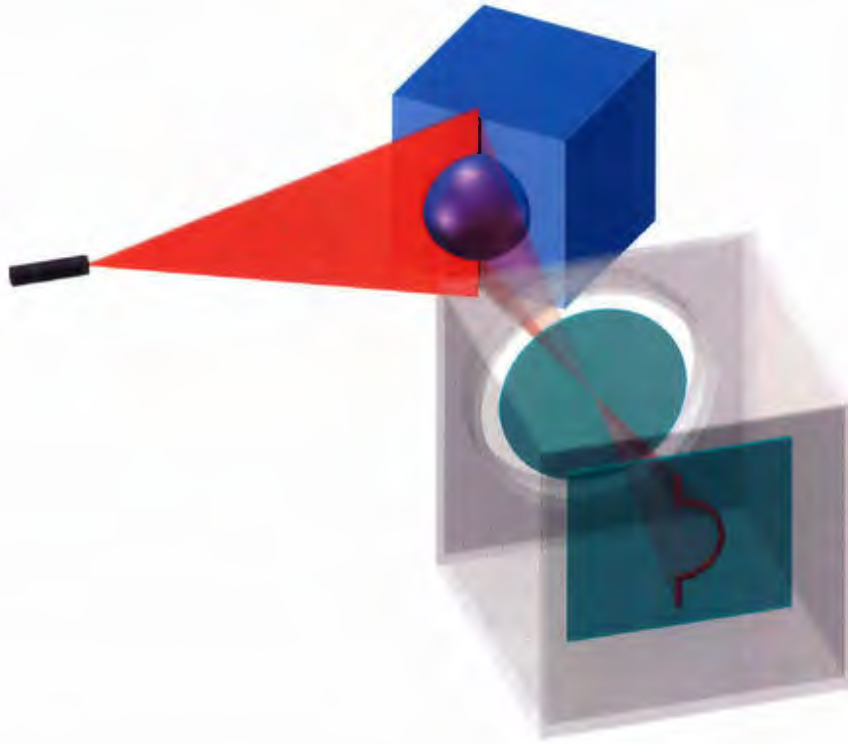


Figure 8.2: Surface Reconstruction System Equipped with Laser Projector

## 8.2 Laser Calibration

To define a plane in 3-Space uniquely the coordinates of at least three non-collinear points lying on that plane must be known. In other words, a line in that plane and a point which is not collinear to that line is sufficient for defining the plane completely. In the past years a number of laser profiler calibration techniques have been presented

in the literature. For example the calibration techniques presented in [51, 56] can be cited. Reference [51] takes an implicit approach towards the laser projector calibration problem where the range is expressed as a function of the location of the laser point on the image plane. Whereas [56] employs the invariant property of cross ratio under perspective projection to obtain a solution. From a practical point of view, the primal challenge is to determine the accurate position of a laser point in the object space. In an effort to address this challenge a novel approach was adopted in this work to calibrate the laser projector.

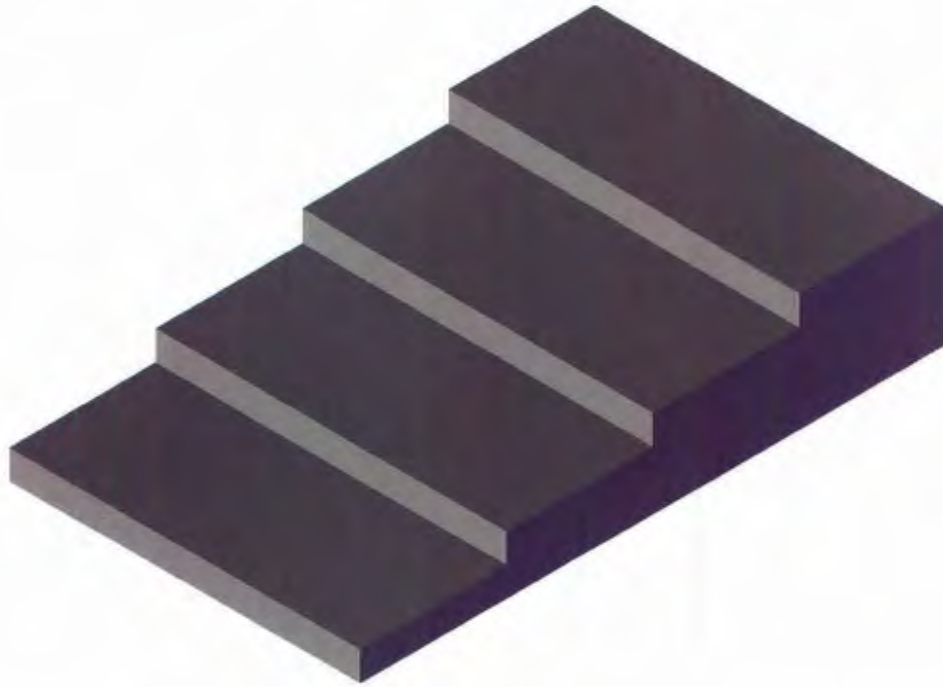


Figure 8.3: Calibration Target for Laser Projector (Step Height 0.25 *inch*)

The geometry and the motion of the calibration rig that was employed to calibrate



the camera offer an opportunity to determine the position of the laser point the object space accurately. The position and the orientation of the calibration table can be determined robustly from the calibrated camera model. In addition, a specially designed laser calibration target (Figure 8.3) can be illuminated by the laser projector to produce the laser points visible to the camera. The calibration target provides a set of planes (in this case, four) which are parallel to the calibration table when the target is mounted on the calibration table. By conventional metrological means the relative positions of these planes with respect to the calibration table can be determined. The extrinsic parameters of the calibrated camera model provide the necessary transformation to obtain the position and the orientation of the planes with respect to the camera coordinate frame. The image coordinates of the laser point can be determined by applying suitable image processing techniques. These image coordinates in turn provides the direction of the light ray associated with the image of the corresponding laser point. The intersection point of the light ray and the parallel planes on the calibration target provides the world coordinates of the laser point with respect to the camera coordinate frame. If one can acquire the positions of at least three such noncollinear laser points in 3-Space, a unique solution can be obtained for the laser calibration problem. Availability of more number of laser points provides an opportunity to solve the problem by least squares analysis. Though in the described setup it is possible to find a unique solution with the help of two parallel planes on the laser calibration target, other two planes provide the additional laser points required to treat the problem as an overdetermined problem of optimization.



### 8.2.1 Mathematical Formulation

Let the common surface normal to the parallel planes on the laser calibration target and the calibration table be  $[p_x p_y p_z]^T$ , expressed in the camera coordinate frame. Hence, the general equation of the parallel planes is provided by Equation 8.4.

$$p_x x + p_y y + p_z z + d_i = 0 \quad (8.4)$$

In Equation 8.4,  $d_i$  is a scalar for  $1 \leq i \leq 4$ . The scalar  $d_i$  can be determined from the geometrical design of the laser calibration target and the extrinsic parameters of the calibrated camera model. Subsequently it can be confirmed by metrological means.

The direction vector  $[\ell_x \ell_y \ell_z]^T$  of the light ray associated with the undistorted image coordinate  $[u v]^T$  of a laser point can be obtained as discussed to Section 8.1. The intersection point of the light ray  $[\ell_x \ell_y \ell_z]^T$  and the plane of the calibration target provides the corresponding world coordinates  $[x_c y_c z_c]^T$  of that laser point. Since this point lies in the laser plane the following equation must hold.

$$n_x x_c + n_y y_c + n_z z_c + n_d = 0 \quad (8.5)$$

Since all the laser points must satisfy Equation 8.5 a system of linear equations can be formed when  $n$  data points are available (Equation 8.6).

$$\begin{bmatrix} x_{c1} & y_{c1} & z_{c1} & 1 \\ \vdots & \vdots & \vdots & \vdots \\ x_{cn} & y_{cn} & z_{cn} & 1 \end{bmatrix} \begin{bmatrix} n_x \\ n_y \\ n_z \\ n_d \end{bmatrix} = \begin{bmatrix} 0 \\ \vdots \\ 0 \end{bmatrix} \quad (8.6)$$

Equation 8.6 is a homogeneous system of linear equations of the form  $\mathbf{A}b = 0$ . The least squares solution that minimizes  $\|\mathbf{A}b\|^2$  is given by the right singular vector of  $\mathbf{A}$

that have the smallest singular value. Singular value decomposition of the coefficient matrix **A** provides the sought after solution.

### 8.3 Localizing the Laser Points

When the laser projector illuminates a planar surface the resultant line exhibits a Gaussian distribution in the lateral direction of the line in the image. Unfortunately the digitization noise and the speckle noise present in the image cause the distribution to be perturbed from the ideal Gaussian curve. The image processing technique, therefore, employs nonlinear least squares analysis to fit a Gaussian curve to the intensity distribution across the laser line in the image.

The Gaussian function in Equation 8.7 is defined by the parameters  $a$ ,  $b$  and  $c$  representing respectively the height of the peak of the Gaussian curve, the location of the mean of the curve and the standard deviation of the curve.

$$G(x) = ae^{-\frac{(x-b)^2}{2c^2}} \quad (8.7)$$

One important property of the Gaussian function is that it approaches zero as the independent variable moves further away from the mean. In the real laser image the gray level intensity distribution across the laser line is offset by some gray cycle value  $\Theta$  from the ideal Gaussian curve (Equation 8.8) in the direction of the dependent variable.

$$G(x) = ae^{-\frac{(x-b)^2}{2c^2}} + \Theta \quad (8.8)$$

Equation 8.8 models the intensity distribution in the real laser image sufficiently. The image processing technique acquires a sample of intensity distribution from each

discrete cross-section of the line. The acquired data is subsequently fitted to the modified Gaussian function in Equation 8.8.

Least squares analysis of a set of data points requires some error criterion on which the analysis operates to minimize that error. The algebraic distance of the analytic Gaussian curve and the actual data point  $(x, y)$  can be one such error criterion.

$$\epsilon = ae^{-\frac{(x-b)^2}{2c^2}} + \Theta - y \quad (8.9)$$

During the actual data mining process this error criterion was found to be more sensitive to noise and hence, performed inconsistently. As a result an alternative error criterion was developed that was more robust to noise.

The tangent on the Gaussian curve at the point  $(x, G(x))$  is represented by Equation 8.10.

$$y = \frac{dG}{dx}x + c_t \quad (8.10)$$

The geometric distance (i.e., perpendicular distance) of the data point  $(x, y)$  from the tangent line is provided by Equation 8.11.

$$d_t = \frac{-\frac{dG}{dx}x + y - c_t}{\sqrt{(\frac{dG}{dx})^2 + 1}} \quad (8.11)$$

Minimization of this geometric distance by nonlinear least squares analysis finds an optimal Gaussian curve that best fits the data. Figure 8.4 demonstrates both the geometric and the algebraic error minimization for a set of data points sampled from a real image.

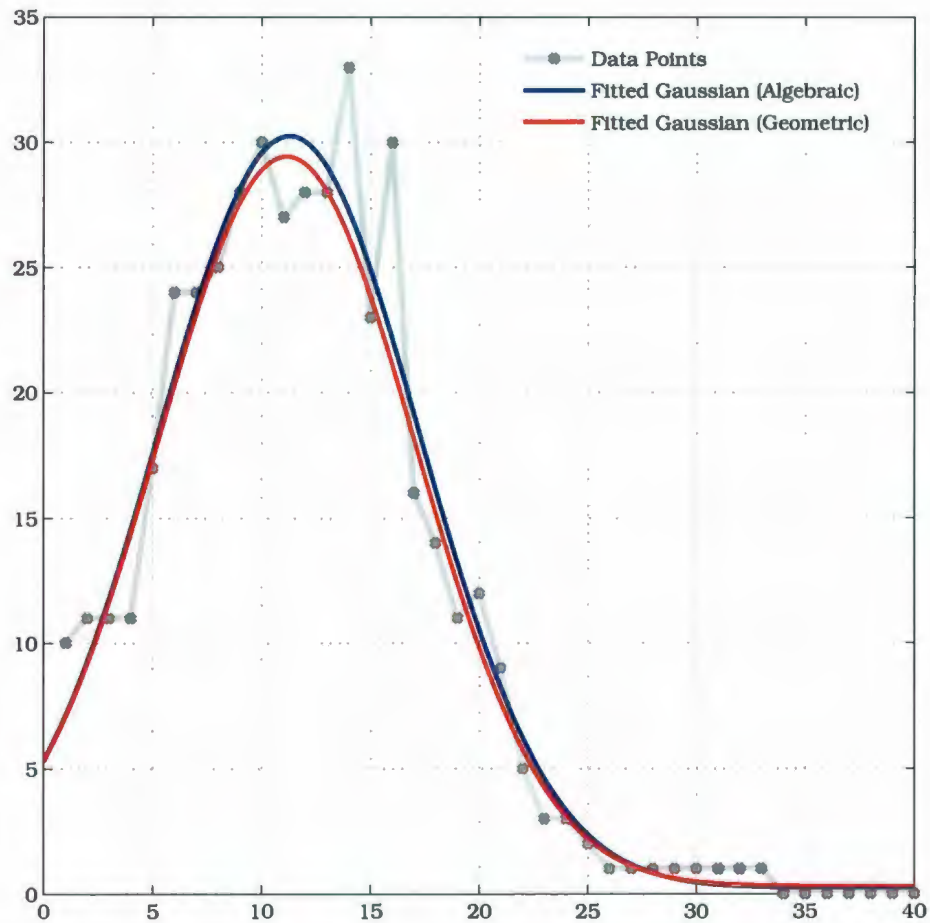


Figure 8.4: Fitting Gaussian Curves to Data Points

### 8.3.1 Initial Approximation

The nonlinear least squares analysis requires good initial approximations of all the unknowns  $a$ ,  $b$ ,  $c$  and  $\Theta$  in order to converge to a solution. Let the given set of data contain  $n$  Cartesian coordinate pairs  $(x_1, y_1), \dots, (x_n, y_n)$ . The initial approximations



for  $\Theta$ ,  $a$  and  $b$  can be calculated as described in Equation 8.12.

$$\begin{aligned}\Theta &= \frac{y_1 + y_n}{2} \\ a &= \max(y) - \Theta \\ b &= \max(x \forall G(x) = a + \Theta)\end{aligned}\tag{8.12}$$

Full width at half maximum (FWHM) of a function is the difference between two extreme values of the independent variable at which the function has half the global maxima of the function. For a Gaussian curve,  $\text{FWHM} \approx 2.3548c$ . From the given data set the FWHM is estimated and subsequently the initial approximation of  $c$  can be provided by the following equation.

$$c = \frac{\text{FWHM}}{2.3548}\tag{8.13}$$

### 8.3.2 Data Reduction

Noise in the measurements, especially in the image space, can result in inaccurate calibration of the laser which is not suitable for applications that demand high accuracy. Hence, the calibration data mined by the image processing technique is preprocessed to discard the potential outliers prior to the actual calibration. Since the intersection of the laser plane and the plane of the calibration target is a line, the data points mined by the image processing technique were fitted to a straight line by least squares analysis. Any data point that had a perpendicular distance from the fitted line greater than some predefined threshold  $v$  was discarded. The rest of the data was fitted to a plane to calibrate the laser projector.

## 8.4 Experimental Surface Reconstruction

The surface of a sphere was reconstructed. The camera and the laser projector were at rest while the sphere was displaced accurately by the calibration rig through a predefined discrete distance to acquire each sample of the spherical surface. The laser illuminated the sphere and the camera captured the image. During the data mining process, the points that did not correspond to the spherical surface were considered unreliable. These points were discarded before the surface reconstruction. The digitization process produced a point cloud. Delaunay triangulation was used to render the point cloud (Figure 8.5). The best fit sphere to the produced data points



Figure 8.5: Reconstruction of A Spherical Surface (Sphere Diameter =  $25.4\text{ mm}$ )

provided a radius of  $12.56\text{ mm}$  whereas the actual radius was  $12.70\text{ mm}$  as confirmed by conventional measurements.

A laser projector calibration technique is presented [57] that employed a digitized spherical surface for accuracy evaluation based on the sphericity error metric

(Equation 8.14).

$$E = \frac{1}{n} \sum_{i=1}^n (r - \hat{r}_i)^2 \quad (8.14)$$

The distance of a point from the center of the best fit sphere is  $\hat{r}_i$  and  $r$  is the actual radius in Equation 8.14. A comparison of the minimum sphericity error that was reported in [57] and the sphericity error resulting from the digitization of the spherical surface is provided in Table 8.1.

Table 8.1: Comparison of Sphericity Error

	[57]	Proposed
Number of Points	19861	3426
Sphericity Error ( $mm^2$ )	0.130	0.043

## Chapter 9

### Conclusion

This thesis endeavored to answer the question whether or not higher accuracy is attainable in a vision based measurement system that employs only off the shelf sensing elements. In our pursuit for an answer, the greatest challenge was identified as the accurate geometrical calibration of the camera, the primary sensor of the surface reconstruction system. A novel camera calibration technique was developed to address this challenge. Performance benchmarking proved this proposed technique to be more robust to the distortion effect of real lens elements. A novel implementation methodology for camera calibration was also proposed. Though the proposed implementation methodology required rigorous effort, it was fairly justified when the improved performance was considered. Finally a novel laser projector calibration technique was developed. The geometric approach that was adopted in the laser calibration technique yielded better results in comparison to the performance of similar systems reported in the literature. This was confirmed by digitizing a spherical surface. The experimental results convincingly showed improvements over the existing



techniques in terms of accuracy.

Because of the time intensive nature of the solution to the reconstruction problem proposed in this work, one can critically remark that the solution is not feasible in an industrial scenario. A superficial review might also validate this argument. But an in depth look reveals that, there are a lot of opportunities to streamline the solution to make it applicable for industrial usage. A list of possible future developments and propositions for streamlining the solution can be found below:

- Experimental calibration of a real fish-eye lens camera can validate the claimed robustness of the proposed camera calibration technique. If the results imply otherwise, the existing framework of the numerical analysis can be used to make necessary modifications.
- The algorithms that were developed for image processing were generally implemented in MATLAB<sup>®</sup>. Higher processing speed is generally guaranteed when the algorithms are implemented in digital hardware or in binary executables.
- In order to minimize the time required for the calibration of the camera specially designed calibration targets can be used. A study focusing on finding the minimum number of point correspondences that is required to calibrate the camera with acceptable accuracy might contribute significantly in minimizing the calibration time.
- In a dynamic reconstruction scenario the laser projector might need to adjust its luminance according to the radiometric nature of the surface. The effect of the dynamic luminance on the calibration parameters of the laser projector can

be studied to confirm whether or not the calibration parameters for the laser projector are dependent on the luminance of the laser source.

- The conventional lens distortion model can be reviewed experimentally with an aim to improve it further.
- The proposed camera calibration technique estimates the optimized camera parameters rather than their true values. Though the calibrated model works fairly well for the calibration scene, in an unstructured environment the calibrated model might fail to produce the required accuracy. It would be worthwhile to identify the camera parameters that have high interaction and to devise an experimental method to estimate them separately.

In addition to these possible future developments, all the aspects of the measurement system can be further evaluated to find better and faster solutions to handle the challenges of close range photogrammetry of the twenty first century. This necessarily means reviewing the entire solution to identify potential bottle-necks, ill conditioning of equations and over parametrization.

This work showed that higher accuracy in a vision based measurement system is definitely attainable, though the effort required for it can be exponentially greater. In contrast to this conclusion, the following quote is worthy of citation:

*"Just how much more closely the numbers can approach the 'true values' depends on our knowledge of the truth. One approaches truth asymptotically, sometimes at the cost of great effort; nevertheless, it is necessary to examine the path toward this ultimate goal and select reasonable limits of achievement."*

- Chester C. Slama, 1980 (*Manual of Photogrammetry*)

# Bibliography

- [1] O. Faugeras, *Three-Dimensional Computer Vision A Geometric View Point*, 1st ed. MIT Press, 1993.
- [2] R. Hartley and A. Zisserman, *Multiple View Geometry in Computer Vision*, 2nd ed. Cambridge University Press, 2004.
- [3] R. Y. Tsai, "A Versatile Camera Calibration Technique for High-Accuracy 3D Machine Vision Based Metrology Using Off-the-Shelf TV Cameras and Lenses," *IEEE Journal of Robotics and Automation*, vol. 3, no. 4, pp. 323-344, 1987.
- [4] J. Heikkila, "Geometric Camera Calibration Using Circular Control Points," *IEEE Transactions On Pattern Analysis And Machine Intelligence*, vol. 22, no. 10, pp. 1066-1077, 2000.
- [5] Z. Zhang, "A Flexible New Technique for Camera Calibration," *IEEE Transactions On Pattern Analysis And Machine Intelligence*, vol. 22, no. 11, pp. 1330-1334, 2000.
- [6] J. Weng, P. Cohen, and M. Herniou, "Camera Calibration with Distortion Models and Accuracy Evaluation," *IEEE Transactions On Pattern Analysis And Machine Intelligence*, vol. 14, no. 10, pp. 965-980, 1992.



- [7] C. C. Slama, C. Theurer, and S. W. Henriksen, Eds., *Manual of Photogrammetry*, 4th ed. American Society of Photogrammetry, 1980.
- [8] J. Kannala and S. S. Brandt, "A generic camera model and calibration method for conventional, wide-angle, and fish-eye lenses," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 28, no. 8, pp. 1335–1340, 2006.
- [9] J. A. Gutierrez and B. S. Armstrong, *Precision Landmark Location for Machine Vision and Photogrammetry Finding and Achieving the Maximum Possible Accuracy*, 1st ed. Springer-Verlag London Limited, 2008, p. 33.
- [10] J. Salvi, X. Armangu, and J. Batlle, "A comparative review of camera calibrating methods with accuracy evaluation," *Pattern Recognition*, vol. 35, no. 7, pp. 1617–1635, 2002.
- [11] F. Remondino and C. Fraser, "Digital Camera Calibration Methods: Considerations and Comparisons," *ISPRS Commission V Symposium 'Image Engineering and Vision Metrology'*, vol. 26, no. 5, pp. 266–272, 2006.
- [12] T. A. Clarke and J. G. Fryer, "The Development of Camera Calibration Methods and Models," *The Photogrammetric Record*, vol. 16, no. 91, pp. 51–66, 1998.
- [13] D. C. Brown, "Close-Range Camera Calibration," *Photogrammetric Engineering & Remote Sensing*, vol. 37, no. 8, pp. 855–866, 1971.
- [14] G.-Q. Wei and S. D. Ma, "Implicit and Explicit Camera Calibration: Theory and Experiments," *IEEE Transactions On Pattern Analysis And Machine Intelligence*, vol. 16, no. 5, pp. 469–480, 1994.



- [15] E. L. Hall, J. B. K. Tio, C. A. McPherson, and F. A. Sadjadi, "Measuring Curved Surfaces for Robot Vision," *Computer*, vol. 15, no. 12, pp. 42–54, Dec 1982.
- [16] B. Triggs, "Autocalibration from planar scenes," in *In Proc. ECCV*, 1998, pp. 89–105.
- [17] Y. Abdel-Aziz and H. Karara, "Direct linear transformation from comparator coordinates into object space coordinates in close-range photogrammetry," *Proceedings of the Symposium on Close-Range Photogrammetry*, pp. 1–18, 1971.
- [18] O. Faugeras and G. Toscani, "The calibration problem for stereo," *Proceedings of the CVPR, IEEE*, pp. 15–20, 1986.
- [19] J. Heikkila and O. Silven, "A Four-step Camera Calibration Procedure with Implicit Image Correction," in *CVPR '97: Proceedings of the 1997 Conference on Computer Vision and Pattern Recognition (CVPR '97)*. Washington, DC, USA: IEEE Computer Society, 1997.
- [20] H. Zollner and R. Sablatnig, "Comparison of Methods for Geometric Camera Calibration using Planar Calibration Targets," in *Digital Imaging in Media and Education, Proceedings of the 28th Workshop of the Austrian Association for Pattern Recognition (OAGM/AAPR)*, vol. 179, 2004, pp. 237–244. [Online]. Available: <http://www.prip.tuwien.ac.at/people/sab/papers/oagm04b.pdf>
- [21] W. Sun and J. R. Cooperstock, "Requirements for Camera Calibration: Must Accuracy Come with a High Price?" *Applications of Computer Vision and the IEEE Workshop on Motion and Video Computing, IEEE Workshop on*, vol. 1, pp. 356–361, 2005.

- [22] X. Feng, M. Cao, H. Wang, and M. Collier, "The Comparison of Camera Calibration Methods Based on Structured-Light Measurement," in *CISP '08: Proceedings of the 2008 Congress on Image and Signal Processing, Vol. 2*. Washington, DC, USA: IEEE Computer Society, 2008, pp. 155–160.
- [23] R. Lenz and R. Tsai, "Techniques for Calibration of the Scale factor and Image Center for High Accuracy 3-D Machine Vision Metrology," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 10, no. 5, pp. 713–720, 1988.
- [24] L. Euler, "Formulae generales pro translatione quacunque corporum rigidorum," *Novi Commentari Academiae Scientiarum Imperialis Petropolitanae*, vol. 20, pp. 189–207, 1775.
- [25] S. B. Kang and S. Member, "Error analysis of pure rotation-based self-calibration," in *In Proc. 8th International Conference on Computer Vision*, 2001, pp. 464–471.
- [26] J. Stuelpnagel, "On the Parametrization of the Three-Dimensional Rotation Group," *SIAM Review*, vol. 6, no. 4, pp. 422–430, Oct., 1964. [Online]. Available: <http://www.jstor.org/stable/2027966>
- [27] J. Schmidt and H. Niemann, "Using Quaternions for Parametrizing 3D Rotations in Unconstrained Nonlinear Optimization," in *Vision, Modeling, and Visualization 2001*. AKA/IOS Press, 2001, pp. 399–406.
- [28] S. L. Altmann, *Rotations, Quaternions, and Double Groups*, 1st ed. Oxford University Press, 1986.

- [29] B. K. Horn, "Tsai's camera calibration method revisited," 2004. [Online]. Available: [http://people.csail.mit.edu/bkph/articles/Tsai\\_Revisited.pdf](http://people.csail.mit.edu/bkph/articles/Tsai_Revisited.pdf)
- [30] B. K. P. Horn, "Closed-form solution of absolute orientation using unit quaternions," *J. Opt. Soc. Am. A*, vol. 4, no. 4, pp. 629–642, 1987.
- [31] N. Krouglicof, "Rigid-Body Pose Measurement from a Single Perspective View," in *Intelligent Autonomous Systems: Proc. of the International Conference IAS-3*, F. C. A. Groen, S. Hirose, and C. E. Thorpe, Eds. Washington: IOS Press, 1993, pp. 368–377.
- [32] C. D. Crane and J. Duffy, *Kinematic Analysis of Robot Manipulators*, 1st ed. The Press Syndicate of The University of Cambridge, 1990, pp. 13–15.
- [33] T. Melen, "Geometrical Modelling and Calibration of Video Cameras for Underwater Navigation," Ph.D. dissertation, Norwegian University of Science and Technology, 1994.
- [34] K. Levenberg, "A Method for the Solution of Certain Non-linear Problems in Least Squares," *Quarterly of Applied Mathematics*, vol. 2, no. 2, pp. 164–168, 1944.
- [35] M. I. A. Lourakis and A. A. Argyros, "Is levenberg-marquardt the most efficient optimization algorithm for implementing bundle adjustment?" in *ICCV '05: Proceedings of the Tenth IEEE International Conference on Computer Vision*. Washington, DC, USA: IEEE Computer Society, 2005, pp. 1526–1531.



- [36] C. Colombo, D. Comanducci, and A. D. Bimbo, *Computer Vision ECCV 2006*. Springer Berlin / Heidelberg, 2006, ch. Camera Calibration with Two Arbitrary Coaxial Circles, pp. 265–276.
- [37] M. Agrawal and L. S. Davis, “Camera calibration using spheres: A semi-definite programming approach,” in *ICCV '03: Proceedings of the Ninth IEEE International Conference on Computer Vision*. Washington, DC, USA: IEEE Computer Society, 2003, p. 782.
- [38] X. Ying and H. Zha, “Linear Approaches to Camera Calibration from Sphere Images or Active Intrinsic Calibration Using Vanishing Points,” in *ICCV '05: Proceedings of the Tenth IEEE International Conference on Computer Vision (ICCV'05) Volume 1*. Washington, DC, USA: IEEE Computer Society, 2005, pp. 596–603.
- [39] K.-Y. K. Wong, P. R. S. Mendonça, and R. Cipolla, “Camera Calibration from Symmetry,” in *Proceedings of the 9th IMA Conference on the Mathematics of Surfaces*. London, UK: Springer-Verlag, 2000, pp. 214–226.
- [40] N. Otsu, “A threshold selection method from gray-level histograms,” *IEEE Transactions on Systems, Man and Cybernetics*, vol. 9, no. 1, pp. 62–66, January 1979.
- [41] A. Fitzgibbon, M. Pilu, and R. B. Fisher, “Direct Least Square Fitting of Ellipses,” *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 21, no. 5, pp. 476–480, 1999.



- [42] V. Pratt, "Direct least-squares fitting of algebraic surfaces," *SIGGRAPH Comput. Graph.*, vol. 21, no. 4, pp. 145–152, 1987.
- [43] R. Halir and J. Flusser, "Numerically Stable Direct Least Squares Fitting of Ellipses," 1998. [Online]. Available: <http://citeseerx.ist.psu.edu/viewdoc/summary?doi=10.1.1.1.7559>
- [44] A. Fitzgibbon, M. Pilu, and R. B. Fisher, "Direct Least Square Fitting of Ellipses," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 21, no. 5, pp. 476–480, 1999.
- [45] T. B. Ay, W. Gander, G. H. Golub, and R. Strebel, "Least-squares fitting of circles and ellipses," *BIT*, vol. 43, pp. 558–578, 1994.
- [46] R. L. Barker, *The Social Work Dictionary*, 5th ed. Washington, DC: NASW Press, 2003, p. 41.
- [47] M. H. Kalos and P. A. Whitlock, *Monte Carlo methods. Vol. 1: Basics*. New York, NY, USA: Wiley-Interscience, 1986, p. 2.
- [48] E. Prados and O. Faugeras, "Shape From Shading," in *Handbook of Mathematical Models in Computer Vision*, Y. C. N. Paragios and O. Faugeras, Eds. Springer, 2006, ch. 23, pp. 375–388. [Online]. Available: <http://perception.inrialpes.fr/Publications/2006/PF06a>
- [49] C. Tomasi and T. Kanade, "Shape and Motion from Image Streams: a Factorization Method," *International Journal of Computer Vision*, Tech. Rep., 1991.

- [50] M. Clerc and S. Mallat, "The Texture Gradient Equation for Recovering Shape from Texture," 2002.
- [51] F. W. DePiero and M. M. Trivedi, "3-D Computer Vision Using Structured Light: Design, Calibration, and Implementation Issues," *Advances in Computers*, vol. 43, pp. 243–278, 1996.
- [52] R. A. Jarvis, "A Perspective on Range Finding Techniques for Computer Vision," *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, vol. PAMI-5, no. 2, pp. 122–139, March 1983.
- [53] P. J. Besl, "Active, optical range imaging sensors," *Machine Vision and Applications*, vol. Volume 1, no. 2, pp. 127–152, 1988.
- [54] L. Zhang, B. Curless, and S. M. Seitz, "Rapid shape acquisition using color structured light and multi-pass dynamic programming," in *In The 1st IEEE International Symposium on 3D Data Processing, Visualization, and Transmission*, 2002, pp. 24–36.
- [55] M. Aldon and O. Strauss, "3d surface segmentation using active sensing," Jun 1991, pp. 1371–1376 vol.2.
- [56] D. Q. Huynh, R. A. Owens, and P. E. Hartmann, "Calibrating a Structured Light Stripe System: A Novel Approach," *Int. J. Comput. Vision*, vol. 33, no. 1, pp. 73–86, 1999.
- [57] K. Yamauchi, H. Saito, and Y. Sato, "Calibration of a structured light system by observing planar object from unknown viewpoints," in *ICPR*, 2008, pp. 1–4.

## Appendix A

# Newton-Gauss Algorithm for Non-Linear Least Squares Analysis

The system of  $n$  nonlinear equations is represented by the vector  $F(\bar{\theta})$  (Equation A.1).

$$F(\bar{\theta}) = \begin{bmatrix} f_1(\bar{\theta}) \\ f_2(\bar{\theta}) \\ f_3(\bar{\theta}) \\ \dots \\ f_n(\bar{\theta}) \end{bmatrix} \quad \text{where, } \bar{\theta} = \begin{bmatrix} \theta_1 \\ \theta_2 \\ \theta_3 \\ \dots \\ \theta_n \end{bmatrix} \quad (\text{A.1})$$

The vector  $\bar{\theta}$  represents the unknown parameters  $\theta_1, \theta_2, \theta_3, \dots, \theta_n$ . First order Taylor series expansion of  $F(\bar{\theta})$  about an initial parameter vector  $\bar{\theta}_0$  is:

$$F(\bar{\theta}) = F(\bar{\theta}_0) + J(\bar{\theta}_0) \times [\bar{\theta} - \bar{\theta}_0] \quad (\text{A.2})$$

Here,  $J(\bar{\theta}_0)$  is the *Jacobian* of  $F(\bar{\theta})$  evaluated at  $\bar{\theta}_0$

The objective is to find the zeros or the roots of  $F(\bar{\theta})$ .

$$\begin{aligned} F(\bar{\theta}) &= 0 \\ \Rightarrow F(\bar{\theta}_0) + J(\bar{\theta}_0) \times [\bar{\theta} - \bar{\theta}_0] &= 0 \end{aligned} \quad (\text{A.3})$$

Substituting  $\beta = (\bar{\theta} - \bar{\theta}_0)$  in Equation A.3 yields the following:

$$F(\bar{\theta}_0) + J(\bar{\theta}_0) \times \beta = 0 \quad (\text{A.4})$$

A practical nonlinear system is usually overdetermined, i.e., number of available data points or equations is more than the unknowns. Moreover, Equation A.4 can not be explicitly satisfied for all data points due to error. Hence, an error term  $\bar{\epsilon}$  will be introduced.

$$F(\bar{\theta}_0) + J(\bar{\theta}_0) \times \beta = \bar{\epsilon} \quad (\text{A.5})$$

The sum of squares of the individual errors is an appropriate error criterion  $q$  which can be readily minimized.

$$\begin{aligned} q &= [\bar{\epsilon}]^T \times [\bar{\epsilon}] \\ &= [F(\bar{\theta}_0) + J(\bar{\theta}_0) \times \beta]^T \times [F(\bar{\theta}_0) + J(\bar{\theta}_0) \times \beta] \end{aligned} \quad (\text{A.6})$$

In order to minimize  $q$  the partial derivative of Equation A.6 with respect to each of the unknown parameters has to be calculated and subsequently has to be set to zero. Taking the partial derivative and subsequent algebraic manipulations yield the following:

$$[J(\bar{\theta}_0)^T \times J(\bar{\theta}_0)] \times \beta = -[J(\bar{\theta}_0)]^T \times [F(\bar{\theta}_0)] \quad (\text{A.7})$$

Equation A.7 is of the linear form,  $\mathbf{Ax} = \mathbf{b}$ . This system solved by *Gaussian Elimination*. To facilitate the Gaussian elimination the matrix,  $[J(\bar{\theta}_0)^T \times J(\bar{\theta}_0)]$ , can be



decomposed by *Choleski Decomposition*. Gaussian elimination of Equation A.7 yields the vector  $\beta$ . A new estimate of the parameter vector  $\bar{\theta}$  can be calculated as follows:

$$\bar{\theta} = \bar{\theta}_0 + \beta \quad (\text{A.8})$$

The algorithm thus continues in an iterative fashion. However, a suitable accuracy criterion is yet to be defined upon reaching which one considers the estimate of the parameter vector is of desired accuracy. Euclidean norm of the correction vector  $\beta$  is a convenient accuracy criterion.

$$\|\beta\| = \sqrt{\beta_1^2 + \beta_2^2 + \cdots + \beta_n^2} \quad (\text{A.9})$$

It is also needed to define an error threshold  $\epsilon_t$ . When  $\epsilon_t > \|\beta\|$  the algorithm terminates iterating.

## Appendix B

### Conversion Between Euler Angles and Quaternions

Let a general rotation is expressed by the Euler angles  $\phi$ ,  $\theta$ , and  $\psi$ , in  $X - Y - Z$  convention. The rotation matrix given by the Euler angles in  $X - Y - Z$  convention is:

$$R = \begin{bmatrix} \cos \theta \cos \psi & -\cos \phi \sin \psi + \sin \phi \sin \theta \cos \psi & \sin \phi \sin \psi + \cos \phi \sin \theta \cos \psi \\ \cos \theta \sin \psi & \cos \phi \cos \psi + \sin \phi \sin \theta \sin \psi & -\sin \phi \cos \psi + \cos \phi \sin \theta \sin \psi \\ -\sin \theta & \sin \phi \cos \theta & \cos \phi \cos \theta \end{bmatrix} \quad (\text{B.1})$$

This general rotation in quaternions can also be expressed as,  $\mathbf{q} = d + a\hat{i} + b\hat{j} + c\hat{k}$ .

The corresponding rotation matrix is given in Equation B.2.

$${}^A R_B = \begin{bmatrix} d^2 + a^2 - b^2 - c^2 & 2ab - 2cd & 2ca + 2bd \\ 2ab + 2cd & d^2 - a^2 + b^2 - c^2 & 2bc - 2ad \\ 2ca - 2bd & 2bc + 2ad & d^2 - a^2 - b^2 + c^2 \end{bmatrix} \quad (\text{B.2})$$

In the following discussion conversion between these two representations of rotation will be derived.

## B.1 Euler Angles to Quaternions

Quaternions corresponding to each Euler angle rotations are:

$$\begin{aligned} \mathbf{q}_x &= \cos \frac{\phi}{2} + \sin \frac{\phi}{2} \hat{i} \\ \mathbf{q}_y &= \cos \frac{\theta}{2} + \sin \frac{\theta}{2} \hat{j} \\ \mathbf{q}_z &= \cos \frac{\psi}{2} + \sin \frac{\psi}{2} \hat{k} \end{aligned} \tag{B.3}$$

By combining these quaternion elements the net rotation given by the Euler angles in quaternion representation is achieved.

$$\mathbf{q} = \mathbf{q}_z \cdot \mathbf{q}_y \cdot \mathbf{q}_x \tag{B.4}$$

## B.2 Quaternions to Euler Angles

This conversion can be derived from the element correspondence in rotation matrices in Equations B.2 and B.1. One can easily derive the following:

$$\begin{aligned} \phi &= \arctan \left( \frac{2bc + 2ab}{d^2 - a^2 - b^2 + c^2} \right) \\ \theta &= \arcsin (2bd - 2ca) \\ \psi &= \arctan \left( \frac{2ab + 2cd}{d^2 + a^2 - b^2 - c^2} \right) \end{aligned} \tag{B.5}$$

## Appendix C

# Rotation Matrix to Quaternions Conversion

The problem is to find the quaternion representing the rotation given by the rotation matrix  $R$ . Let  $r_{ij}$  be the element of the rotation matrix  $R$  in the  $i^{\text{th}}$  row and the  $j^{\text{th}}$  column. The rotation matrix corresponding to the quaternion  $\mathbf{q} = d + a\hat{i} + b\hat{j} + c\hat{k}$  is given in Equation B.2. One can easily show that:

$$d = \frac{1}{2}\sqrt{1 + r_{11} + r_{22} + r_{33}} \quad (\text{C.1})$$

Other components of the quaternion can be calculated by the following formulae:

$$a = \frac{r_{32} - r_{23}}{4d} \quad (\text{C.2})$$

$$b = \frac{r_{13} - r_{31}}{4d} \quad (\text{C.3})$$

$$c = \frac{r_{21} - r_{12}}{4d} \quad (\text{C.4})$$









